

A Meshless Approach to Capturing Moving Interfaces in Passive Transport Problems

L. Mai-Cao* and T. Tran-Cong[†]

October 21, 2008

Abstract

This paper presents a new meshless numerical approach to solving a special class of moving interface problems known as the passive transport where an ambient flow characterized by its velocity field causes the interfaces to move and deform without any influences back on the flow. In the present approach, the moving interface is captured by the level set method at all time as the zero contour of a smooth function known as the level set function whereas one of the two new meshless schemes, namely the SL-IRBFN based on the semi-Lagrangian method and the Taylor-IRBFN scheme based on Taylor series expansion, is used to solve a convective transport equation for advancing the level set function in time. In addition, a mass correction is introduced after the reinitialization step to ensure mass conservation. Some basic tests are preformed to verify the accuracy and stability of the new numerical schemes which are then applied to simulate bubbles moving, stretching and merging in an ambient flow to demonstrate the performance of the new meshless approach.

Keyword: Level set method, meshless method, radial basis functions, IRBFN, moving interfaces, passive transport, Taylor-IRBFN, Semi-Lagrangian approach.

1 Introduction

Numerical methods for moving interface problems in general, or passive transport problems in particular, have been increasingly studied in recent years. A moving interface is defined as the time-dependent boundary $\Gamma(t)$ of $\Omega \subset \mathbf{R}^d, d = 1, 2, 3$, that has an outward unit normal \mathbf{n} and a normal velocity (also known as speed) F at each point. In a passive transport problem one would like to determine the evolution of $\Gamma(t)$ with time driven by a given externally generated velocity field \mathbf{v} such that $F = \mathbf{v} \cdot \mathbf{n}$. In such a problem, the influence of the moving interfaces on the velocity field is ignored. When it moves, the interface $\Gamma(t)$ might undergo topological changes, i.e. splitting of an interface and/or merging of several interfaces.

There are two basic approaches to modeling the motion of the interfaces: moving-grid and fixed-grid methods. In the moving-grid methods, the interface is treated as the boundary of a moving surface-fitted grid [Floryan and Rasmussen (1989)]. This approach allows a precise representation of the interface whereas its main drawback is the severe deformation of the mesh due to the interface motion. The second approach, which is based on fixed grids, includes tracking and capturing methods. The tracking methods explicitly represent the moving interface by means of predefined markers [Unverdi and Tryggvason (1992)]. In capturing methods, on the other hand, the moving interface is not explicitly tracked, but rather captured via a characteristic function. Examples of

*Faculty of Geology and Petroleum Engineering, Ho Chi Minh City University of Technology, 268 Ly Thuong Kiet St, District 10, Ho chi Minh City, Vietnam

[†]Corresponding author. Faculty of Engineering and Surveying, University of Southern Queensland, Toowoomba Qld 4350, Australia, email: trancong@usq.edu.au, Fax: +61 7 46312526.

the capturing methods are phase field method [Jacqmin (1999)], volume-of-fluid method [Hirt and Nichols (1981)] and level set method [Osher and Sethian (1988)]. The characteristic function used to implicitly describe the moving interface is the order parameter in the phase field method, volume fraction in the volume-of-fluid method and level set function in the level set method. For these (capturing) methods, no grid manipulation (e.g. rezoning/remeshing) is needed to maintain the overall accuracy even when the interface undergoes large deformation. In this work, the level set method is used to capture the moving interfaces.

The underlying idea of the level set method is to embed the moving interface $\Gamma(t)$ as the zero level set of a smooth (at least Lipschitz continuous) function $\phi(\mathbf{x}, t)$ known as the level set function [Osher and Sethian (1988)]. The moving interface can be then captured at any time by locating the set of $\Gamma(t)$ for which $\phi(\mathbf{x}, t)$ vanishes. The level set function is advanced with time by a convective transport equation known as the level set equation. Usually, $\phi(\mathbf{x}, t)$ is initialized as a signed distance function to the interface [Sethian (1999); Osher and Fedkiw (2003)]. Due to numerical error, however, this feature is not necessarily held. Reinitialization is therefore needed to make the level set function signed distance function after certain time steps, which could be achieved by solving a time-dependent PDE to steady state [Sussman et al. (1994)]. It has been reported that such a reinitialization procedure could introduce some numerical diffusion which results in an inaccuracy of the interface location and some loss of mass [Tornberg (2000)]. The procedure has been improved in [Chang et al. (1996); Sussman and Fatemi (1999); Peng et al. (1999)]. In this work, an additional mass correction based on a well-known formula for the first variation of a volume integral [Cuvelier and Schulkes (1990)] is introduced after the reinitialization step to prevent any significant losses of mass.

The level set method has been applied widely in fluid dynamics [Sussman et al. (1994); Sussman and Smereka (1997); Iafrati et al. (2001)], and structural shape and topology optimization [Wang et al. (2007)], to name just a few applications. Some conservative schemes have been used to solve the level set equation such as Lax-Friedrichs [Crandall and Lions (1984)], Essentially Non-Oscillatory ENO [Shu and Osher (1989)], and Godunov's schemes [Bardi and Osher (1991)]. In this paper, two new meshless numerical schemes, namely the SL-IRBFN scheme based on the semi-Lagrangian method and the Taylor-IRBFN motivated by the well-known Taylor-Galerkin method, are proposed to deal with the level set equation. In contrast to the meshless local Petrov-Galerkin (MLPG) method [Atluri (2004)], the present approach, also truly meshless, is based on global radial basis function network approximants. Unlike the traditional differential approach [Kansa (1990); Sarler (2005); Shu et al. (2005)], the present method is based on integrated (indirect, integral) radial basis function networks (IRBFN) [Mai-Duy and Tran-Cong (2001a); Mai-Duy and Tran-Cong (2001b); Mai-Duy and Tran-Cong (2003); Mai-Duy (2004); Mai-Cao and Tran-Cong (2005)].

The semi-Lagrangian method can be considered as a hybrid approach between the Eulerian and the Lagrangian methods [Staniforth and Cote (1991)]. An Eulerian scheme retains the regularity of the mesh but requires small time steps in order to maintain stability. A Lagrangian scheme, on the other hand, is less restricted by stability requirements and allows larger time steps. However, since the fluid particles, initially regularly spaced, move with time, they usually become irregularly spaced as the system evolves. The semi-Lagrangian advection scheme combines the advantages of both schemes - the regularity of the Eulerian scheme and the enhanced stability of the Lagrangian scheme. The basic idea is to discretize the Lagrangian derivative of the transport quantity in time instead of the Eulerian derivative. It involves backward time integration along the characteristic curve to find the departure point of a fluid particle arriving at an Eulerian grid point. The solution at the departure points is then obtained by interpolation. Interested readers are referred to [Staniforth and Cote (1991); Oliveira and Baptista (1995); Behrens and Iske (2002)] and the references therein for details on semi-Lagrangian methods.

Another well-known numerical method, namely the Taylor-Galerkin method, is widely used for solving convective transport equations [Donea (1984)]. This method is based on the Taylor series expansion about a point in time of a function including higher-order time derivatives. In general, by replacing the temporal derivatives in the Taylor series expansion with the corresponding spatial ones via the differential equation to be solved, the accuracy and the stability of the numerical solution can be improved [Donea and Huerta (2003)]. For the Taylor-Galerkin method, the resultant semi-discrete equation is discretized in space using the standard Galerkin FEM method. For the Taylor-IRBFN scheme, on the other hand, the IRBFN method is used for spatial discretization.

The remaining of this paper is organized as follows. Firstly the IRBFN method, the level set method, the meshless semi-Lagrangian SL-IRBFN and the Taylor-IRBFN schemes are introduced.

The new meshless approach to solving passive transport problems is then presented with a detailed discussion on all “ingredients” mentioned above, particularly the SL-IRBFN, Taylor-IRBFN schemes and the additional mass correction procedure. Finally, the individual schemes and the new approach are verified with some basic tests and demonstrated with some typical passive transport problems.

2 IRBFN approximation of functions and its derivatives

Let $u(\mathbf{x}, t)$ be an unknown function continuously defined on $Q_T := (0, T) \times \Omega$, where $\Omega \subset \mathbf{R}^d$, $d = 1, 2, 3$ is a bounded domain. For convenience, the components of a typical point are denoted by $\mathbf{x} = (x, y, z)$ and typically derivatives with respect to x are used to illustrate the derivation of the method. Given a set of discrete points $\{\mathbf{x}_j\}_{j=1}^M$ in Ω and the corresponding nodal values of the function at certain point in time t , $\mathbf{u}(t) = [u_1(t), u_2(t), \dots, u_M(t)]^T$, the IRBFN formulation for the approximation of the function and its derivatives (e.g. with respect to x) is written as follows.

$$\frac{\partial^2 u(\mathbf{x}, t)}{\partial x^2} \approx \hat{\mathbf{g}}(\mathbf{x})^T \mathbf{G}^{-1} \mathbf{u}(t), \quad (1)$$

$$\frac{\partial u(\mathbf{x}, t)}{\partial x} \approx \tilde{\mathbf{g}}(\mathbf{x})^T \mathbf{G}^{-1} \mathbf{u}(t), \quad (2)$$

$$u(\mathbf{x}, t) \approx \mathbf{g}(\mathbf{x})^T \mathbf{G}^{-1} \mathbf{u}(t), \quad (3)$$

where $\hat{\mathbf{g}}(\mathbf{x})$ is a set of basis functions whose j^{th} component is defined as follows.

$$\begin{aligned} \hat{g}_j(\mathbf{x}) &= \varphi(\|\mathbf{x} - \mathbf{x}_j\|), \quad j = 1, \dots, N, \\ \hat{g}_j(\mathbf{x}) &= 0, \quad j = N + 1, \dots, \bar{N}, \end{aligned} \quad (4)$$

in which $\varphi(\|\mathbf{x} - \mathbf{x}_j\|)$ are radial basis functions such as Hardy’s multiquadrics

$$\varphi(\|\mathbf{x} - \mathbf{x}_j\|) = \sqrt{r_j^2 + s_j^2}, \quad j = 1, \dots, N, \quad (5)$$

or Duchon’s thin plate splines

$$\varphi(\|\mathbf{x} - \mathbf{x}_j\|) = r_j^{2m} \log r_j, \quad j = 1, \dots, N, \quad (6)$$

where m is the TPS order, $r_j = \|\mathbf{x} - \mathbf{x}_j\|$ is the Euclidian norm, and s_j is the RBF shape parameter given by [Moody and Darken (1989)]

$$s_j = \beta d_j^{min}, \quad (7)$$

in which β is the user-defined parameter and d_j^{min} is the distance from the j^{th} data point to its nearest neighboring point.

The functions $\tilde{\mathbf{g}}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$ in (2) and (3) are obtained by symbolically integrating $\hat{\mathbf{g}}(\mathbf{x})$ in the x direction once and twice, respectively. The matrix \mathbf{G} is defined as

$$\mathbf{G} = \begin{bmatrix} g_1(\mathbf{x}_1) & g_2(\mathbf{x}_1) & \dots & g_{\bar{N}}(\mathbf{x}_1) \\ g_1(\mathbf{x}_2) & g_2(\mathbf{x}_2) & \dots & g_{\bar{N}}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ g_1(\mathbf{x}_M) & g_2(\mathbf{x}_M) & \dots & g_{\bar{N}}(\mathbf{x}_M) \end{bmatrix}, \quad (8)$$

where $g_j(\mathbf{x}), j = 1, \dots, \bar{N}$ is the j^{th} component of $\mathbf{g}(\mathbf{x})$, and $\bar{N} = N + P$ in which P is the number of discrete points needed to approximate the constants of integration. Details on the derivation of the IRBFN formulation and numerical investigations of the IRBFN method can be found in [Mai-Cao and Tran-Cong (2005)] for transient problems and in [Mai-Duy et al. (2007)] for fluid flow applications.

For a more compact form, the IRBFN formulation can be written as follows.

$$u_{xx}(\mathbf{x}, t) \equiv \frac{\partial^2 u(\mathbf{x}, t)}{\partial x^2} \approx \psi_{\partial_{xx}}(\mathbf{x})^T \mathbf{u}(t), \quad (9)$$

$$u_x(\mathbf{x}, t) \equiv \frac{\partial u(\mathbf{x}, t)}{\partial x} \approx \psi_{\partial_x}(\mathbf{x})^T \mathbf{u}(t), \quad (10)$$

$$u(\mathbf{x}, t) \approx \psi(\mathbf{x})^T \mathbf{u}(t), \quad (11)$$

where

$$\psi_{\partial_{xx}}(\mathbf{x}) = \hat{\mathbf{g}}(\mathbf{x})^T \mathbf{G}^{-1}, \quad (12)$$

$$\psi_{\partial_x}(\mathbf{x}) = \tilde{\mathbf{g}}(\mathbf{x})^T \mathbf{G}^{-1}, \quad (13)$$

$$\psi(\mathbf{x}) = \mathbf{g}(\mathbf{x})^T \mathbf{G}^{-1}. \quad (14)$$

Let \mathcal{S} be a certain differential operator in space that operates on the scalar function $u(\mathbf{x}, t)$ in $\Omega \in \mathbf{R}^d$, $d = 1, 2, 3$, the IRBFN formulation above can be then rewritten in a generic form for approximating function $u(\mathbf{x}, t)$ and/or its derivatives as follows.

$$\mathcal{S}u(\mathbf{x}, t) \approx \psi_{\mathcal{S}}^T(\mathbf{x}) \mathbf{u}(t), \quad (15)$$

where $\psi_{\mathcal{S}}(\mathbf{x})$ is the vector whose components are the results of the application of operator \mathcal{S} on the corresponding components of $\psi(\mathbf{x})$,

$$\psi_{\mathcal{S}}(\mathbf{x}) = [\mathcal{S}\psi_1(\mathbf{x}), \mathcal{S}\psi_2(\mathbf{x}), \dots, \mathcal{S}\psi_M(\mathbf{x})]^T. \quad (16)$$

For a special case where \mathcal{S} is the identity operator, $\mathcal{S} = \mathcal{I}$, one gets the approximation of function $u(\mathbf{x}, t)$. Otherwise, one obtains the corresponding derivative of the function. For example, if $\mathcal{S} = \frac{\partial}{\partial y} \equiv \partial_y$, one has the approximation of the first order derivative of $u(x, y, t)$ in the y direction as follows.

$$\mathcal{S}u(x, y, t) = \frac{\partial}{\partial y} u(x, y, t) \approx \psi_{\partial_y}^T(x, y) \mathbf{u}(t). \quad (17)$$

3 Level set method

In the level set method, the moving interface $\Gamma(t)$ which bounds an open region $\Omega \subset \mathbf{R}^d$ ($d = 2, 3$) is embedded as the zero level set of a higher dimensional function $\phi(\mathbf{x}, t)$ such that

$$\Gamma(t) = \{\mathbf{x} \in \mathbf{R}^d \mid \phi(\mathbf{x}, t) = 0\}$$

Initially, ϕ is defined as the signed distance function from the front such that

$$\phi(\mathbf{x}, t) = \begin{cases} +d(\mathbf{x}, t) & \mathbf{x} \in \Omega^+ \\ 0 & \mathbf{x} \in \Gamma \\ -d(\mathbf{x}, t) & \mathbf{x} \in \Omega^- \end{cases} \quad (18)$$

where $d(\mathbf{x}, t)$ represents the Euclidean distance from \mathbf{x} to the interface, Ω^- and Ω^+ are interior and exterior regions respectively. The interface can be then captured at any time by locating the set of $\Gamma(t)$ for which ϕ vanishes. In other words, instead of working with the interface, one evolves the level set with the following transport equation for ϕ ,

$$\phi_t + \mathbf{v} \cdot \nabla \phi = 0, \quad \phi(\mathbf{x}, t = 0) = \phi_0(\mathbf{x}), \quad (19)$$

where $\phi_0(\mathbf{x})$ is a given function. Whenever needed, the moving interface can be extracted as the zero level of the level set function $\phi(\mathbf{x}, t)$. It is noted that while the level set function $\phi(\mathbf{x}, t)$ is initialized as a signed distance function from the free surface, this is not necessarily true as time proceeds. In order to keep the numerical solution accurate, one needs to reinitialize $\phi(\mathbf{x}, t)$ to be the signed distance function from the evolving front $\Gamma(t)$ after certain period in time. This is accomplished by solving the following problem to steady state:

$$\phi_t = S_\epsilon(\bar{\phi})(1 - |\nabla \phi|), \quad \phi(x, y, t = 0) = \bar{\phi}(x, y) \quad (20)$$

where S_ϵ denotes the smoothed sign function

$$S_\epsilon(\bar{\phi}) = \frac{\bar{\phi}}{\sqrt{\bar{\phi}^2 + \epsilon^2}} \quad (21)$$

in which ϵ can be chosen to be the minimum distance from any data point to the others.

As mentioned earlier, due to numerical diffusion coming from the approximation of $\text{sign}(\phi)$ in solving equation (20), the reinitialization procedure presented above could move the interface location and cause some losses of mass. A mass correction which is added after the reinitialization step is described in section §6.4.

4 SL-IRBFN scheme for convective transport equations

Consider the transport equation with source term

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = f(\mathbf{x}, t), \quad (22)$$

where $\phi = \phi(\mathbf{x}, t)$ is a scalar quantity, and $\mathbf{v}(\mathbf{x})$ is a given convection velocity. The above equation can be written in the following Lagrangian form

$$\frac{d\phi}{dt} = f(\mathbf{x}, t) \quad (23)$$

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}(\mathbf{x}, t) \quad (24)$$

The procedure for solving equations (23) and (24) using semi-Lagrangian method is as follows.

- At each time step, track backward particles that arrive at the grid points over a single time step along characteristic curves (24) to their departure points;
- Compute the solution values at the departure points;
- Solve (23) for the current time step using the solution values at the departure points as the initial values;
- Advance to the next time step and repeat the above steps until the predefined time is reached.

Applying the semi-Lagrangian method to problem (23) in which the first-order backward Euler difference scheme is used for the time derivative, one obtains

$$\frac{\phi^{n+1} - \phi_d^n}{\Delta t} = f^{n+1}, \quad (25)$$

where ϕ_d^n is the value of ϕ at the departure points \mathbf{x}_d . ϕ_d^n is determined by first solving the following equation

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}(\mathbf{x}, t), \quad \mathbf{x}^{n+1} = \mathbf{x}_a, \quad (26)$$

backward in one single step for the departure points \mathbf{x}_d at time t^n with the initial condition $\mathbf{x}^{n+1} = \mathbf{x}_a$. ϕ_d^n can be then obtained via interpolation from ϕ^n at grid points. In the above equation, \mathbf{x}_a is the position of the arrival points which are the grid points at time t^{n+1} . Equation (26) can be solved by the explicit midpoint rule [Temperton and Staniforth (1987)] as follows.

$$\hat{\mathbf{x}} = \mathbf{x}_a - \frac{\Delta t}{2} \mathbf{v}(\mathbf{x}_a, t^n), \quad (27)$$

$$\mathbf{x}_d = \mathbf{x}_a - \Delta t \mathbf{v} \left(\hat{\mathbf{x}}, t^n + \frac{\Delta t}{2} \right). \quad (28)$$

By letting

$$\delta = \mathbf{x}_a - \mathbf{x}_d, \quad (29)$$

and substituting (27),(29) into (28), one obtains

$$\delta = \Delta t \mathbf{v} \left(\mathbf{x}_a - \frac{\Delta t}{2} \mathbf{v}(\mathbf{x}_a, t^n), t^n + \frac{\Delta t}{2} \right). \quad (30)$$

Once equation (30) is solved for δ , the departure points \mathbf{x}_d can be found via equation (29). It is noted that the velocity field at $t = t^n + \frac{\Delta t}{2}$ in (30) can be determined by extrapolation using the Adams-Bashforth formula

$$\mathbf{v}(\mathbf{x}, t^n + \frac{\Delta t}{2}) = \frac{3}{2} \mathbf{v}(\mathbf{x}, t^n) - \frac{1}{2} \mathbf{v}(\mathbf{x}, t^n - \Delta t) + \mathcal{O}(\Delta t^2). \quad (31)$$

Alternatively, equation (26) can be solved by an implicit midpoint rule as follows.

$$\hat{\mathbf{x}} = \mathbf{x}_a - \frac{\Delta t}{2} \mathbf{v} \left(\hat{\mathbf{x}}, t^n + \frac{\Delta t}{2} \right). \quad (32)$$

$$\mathbf{x}_d = \mathbf{x}_a - \Delta t \mathbf{v} \left(\hat{\mathbf{x}}, t^n + \frac{\Delta t}{2} \right). \quad (33)$$

In this case, one has to solve the following equation for δ ,

$$\delta = \Delta t \mathbf{v} \left(\mathbf{x}_a - \frac{\delta}{2}, t^n + \frac{\Delta t}{2} \right). \quad (34)$$

Although equation (34) has to be solved iteratively, it converges after just a few iterations provided that $\max |\nabla \mathbf{v}| \Delta t$ is sufficiently small [Allievi and Bermejo (2000)]. To enhance the accuracy of the integration, higher-order methods should be used. In this work, the IRBFN semi-discrete scheme, presented in [Mai-Cao and Tran-Cong (2005)] with the fourth-order Runge-Kutta method, is used. In general, the departure points \mathbf{x}_d do not coincide with the grid points. The values of ϕ_d^n at those points are then obtained by interpolation. The IRBFN method with Duchon TPS basis functions is used for this purpose. Alternatively, the cubic spline interpolation can be used. After getting ϕ_d^n , the new value ϕ^{n+1} at time t^{n+1} can be obtained by equation (25). For convective transport equation with no source term, the semi-Lagrangian scheme reduces to $\phi^{n+1} = \phi_d^n$, i.e. the value of function ϕ remains constant on the characteristics and the old value is simply copied into its new position on the regular grid.

5 Taylor-IRBFN schemes for convective transport equations

Consider the two-dimensional pure convective transport equation

$$\frac{\partial \phi}{\partial t} + u \frac{\partial \phi}{\partial x} + v \frac{\partial \phi}{\partial y} = 0, \quad (35)$$

$$\phi(x, y, t = 0) = \phi^0(x, y), \quad (36)$$

where $\phi^0(x, y)$ is a given function, and $u = u(x, y)$ and $v = v(x, y)$ are the component of a given time-independent velocity field in x and y direction, respectively. For the sake of presentation using the same notations as in [Donea (1984)], equation (35) is rewritten as follows.

$$\phi_t = -u\phi_x - v\phi_y, \quad (37)$$

where ϕ_t , ϕ_x , ϕ_y are the derivatives of ϕ in time, x and y direction, respectively.

In the remaining parts of this section, two formulations of the Taylor-IRBFN scheme, namely TE-IRBFN and TCN-IRBFN are derived to solve the problem under consideration. The two formulations differ in the manner the first-order derivative in time is approximated. The former is based on the Euler difference formula whereas the Crank-Nicolson method is used in the latter.

5.1 The TE-IRBFN Scheme

The TE-IRBFN scheme for solving (35) is derived as follows.

Firstly, applying Taylor series expansion of ϕ forward about $t = t^n$ yields

$$\phi^{n+1} = \phi^n + \Delta t \phi_t^n + \frac{\Delta t^2}{2} \phi_{tt}^n + \frac{\Delta t^3}{6} \phi_{ttt}^n + \mathcal{O}(\Delta t^4), \quad (38)$$

or

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = \phi_t^n + \frac{\Delta t}{2} \phi_{tt}^n + \frac{\Delta t^2}{6} \phi_{ttt}^n + \mathcal{O}(\Delta t^3). \quad (39)$$

Secondly, by differentiating equation (37) successively up to the third order derivative in time and replacing the first order derivatives in time with the corresponding spatial derivatives in the right-hand side of the transport equation to be solved (37), one obtains

$$\phi_{tt} = [(u u_x + u_y v) \partial_x + u^2 \partial_{xx} + (u v_x + v v_y) \partial_y + v^2 \partial_{yy} + 2uv \partial_{xy}] \phi, \quad (40)$$

$$\phi_{ttt} = [(uu_x + u_yv)\partial_x + u^2\partial_{xx} + (uv_x + vv_y)\partial_y + v^2\partial_{yy} + 2uv\partial_{xy}] \phi_t, \quad (41)$$

where ∂_x and ∂_y denote the spatial differential operators in x and y directions, respectively. The last first-order time derivative in equation (41) is kept to avoid high-order spatial derivatives in the resultant formulas.

By using the new differential operator notation ∂_χ defined as

$$\partial_\chi = [(uu_x + u_yv)\partial_x + u^2\partial_{xx} + (uv_x + vv_y)\partial_y + v^2\partial_{yy} + 2uv\partial_{xy}], \quad (42)$$

one has the simpler forms of equations (40) and (41) as follows.

$$\phi_{tt} = \partial_\chi \phi, \quad (43)$$

$$\phi_{ttt} = \partial_\chi \phi_t. \quad (44)$$

Next, substituting (37), (43) and (44) into (39) yields

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = (-u\partial_x - v\partial_y) \phi^n + \frac{\Delta t}{2} \partial_\chi \phi^n + \frac{\Delta t^2}{6} \partial_\chi \phi_t^n + \mathcal{O}(\Delta t^3). \quad (45)$$

Finally, by replacing ϕ_t^n in the above equation with the Euler difference formula

$$\phi_t^n = \frac{\phi^{n+1} - \phi^n}{\Delta t}, \quad (46)$$

and rearranging the terms, one obtains

$$\left(1 - \frac{\Delta t^2}{6} \partial_\chi\right) \Delta \phi = \left[\Delta t (-u\partial_x - v\partial_y) + \frac{\Delta t^2}{2} \partial_\chi\right] \phi^n + \mathcal{O}(\Delta t^3), \quad (47)$$

where $\Delta \phi = \phi^{n+1} - \phi^n$. For each time step, equation (47) is solved for $\Delta \phi$, and the solution at $t = t^{n+1}$ is obtained via $\phi^{n+1} = \phi^n + \Delta \phi$.

Using the IRBFN method for the spatial discretization of equation (47), the fully discrete TE-IRBFN formulation for problem (35) can be then derived as follows.

$$\begin{aligned} & \left[1 - \frac{\Delta t^2}{6} \psi_{\partial_\chi}^T(\mathbf{x}_i)\right] \Delta \phi = \\ & \left[\Delta t \left(-u_i \psi_{\partial_x}^T(\mathbf{x}_i) - v_i \psi_{\partial_y}^T(\mathbf{x}_i)\right) + \frac{\Delta t^2}{2} \psi_{\partial_\chi}^T(\mathbf{x}_i)\right] \phi^n, \\ & i = 1, 2, \dots, M, \end{aligned} \quad (48)$$

where ψ_{∂_χ} , ψ_{∂_x} and ψ_{∂_y} are the IRBFN approximations to the differential operator ∂_χ , ∂_x and ∂_y , respectively; u_i and v_i are the components of the velocity field \mathbf{v} at position \mathbf{x}_i in x and y directions, respectively.

5.2 The TCN-IRBFN Scheme

The TCN-IRBFN scheme for solving (35) is derived as follows.

Firstly, applying Taylor series expansions of function ϕ forward about $t = t^n$ and backward about $t = t^{n+1}$ yields

$$\phi^{n+1} = \phi^n + \Delta t \phi_t^n + \frac{\Delta t^2}{2} \phi_{tt}^n + \frac{\Delta t^3}{6} \phi_{ttt}^n + \mathcal{O}(\Delta t^4), \quad (49)$$

$$\phi^n = \phi^{n+1} - \Delta t \phi_t^{n+1} + \frac{\Delta t^2}{2} \phi_{tt}^{n+1} - \frac{\Delta t^3}{6} \phi_{ttt}^{n+1} + \mathcal{O}(\Delta t^4) \quad (50)$$

Subtracting (50) from (49) and rearranging terms result in

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = \frac{(\phi_t^n + \phi_t^{n+1})}{2} + \frac{\Delta t}{4}(\phi_{tt}^n - \phi_{tt}^{n+1}) + \frac{\Delta t^2}{12}(\phi_{ttt}^n + \phi_{ttt}^{n+1}) + \mathcal{O}(\Delta t^3). \quad (51)$$

Next, substituting (37),(40),(41) into (51) and rearranging terms, one obtains

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = -\frac{1}{2}(u\partial_x + v\partial_y)(\phi^n + \phi^{n+1}) + \frac{\Delta t}{4}\partial_\chi(\phi^n - \phi^{n+1}) + \frac{\Delta t^2}{12}\partial_\chi(\phi_t^n + \phi_t^{n+1}) + \mathcal{O}(\Delta t^3), \quad (52)$$

where the differential operator notation ∂_χ is defined in (42).

Finally, by applying the Crank-Nicolson time-stepping [Donea (1984)]

$$\frac{1}{2}(\phi_t^n + \phi_t^{n+1}) = \frac{\phi^{n+1} - \phi^n}{\Delta t}, \quad (53)$$

one obtains the semi-discrete form of (37) as follows.

$$\left[1 + \frac{\Delta t}{2}(u\partial_x + v\partial_y) + \frac{\Delta t^2}{12}\partial_\chi\right] \Delta\phi = -\Delta t(u\partial_x + v\partial_y)\phi^n, \quad (54)$$

where $\Delta\phi = \phi^{n+1} - \phi^n$. For each time step, equation (54) is solved for $\Delta\phi$, and the solution at $t = t^{n+1}$ is obtained via $\phi^{n+1} = \phi^n + \Delta\phi$.

Using the IRBFN method for the spatial discretization of equation (54), the fully discrete TCN-IRBFN formulation for problem (35) can be then derived as follows.

$$\left\{1 + \frac{\Delta t}{2} \left[u_i \psi_{\partial_x}^T(\mathbf{x}_i) + v_i \psi_{\partial_y}^T(\mathbf{x}_i) \right] + \frac{\Delta t^2}{12} \psi_{\partial_\chi}^T(\mathbf{x}_i) \right\} \Delta\phi = -\Delta t \left[u_i \psi_{\partial_x}^T(\mathbf{x}_i) + v_i \psi_{\partial_y}^T(\mathbf{x}_i) \right] \phi^n, \quad i = 1, \dots, M, \quad (55)$$

where ψ_{∂_χ} , ψ_{∂_x} and ψ_{∂_y} are the IRBFN approximations to the differential operator ∂_χ , ∂_x and ∂_y , respectively; u_i and v_i are the components of the velocity field \mathbf{v} at position \mathbf{x}_i in x and y directions, respectively.

6 A new meshless approach to passive transport problems

The present new meshless numerical approach to capturing moving interfaces in passive transport problems is built by bringing all ingredients previously presented together and consists of the following steps.

Step 1: Initialize the level set function $\phi(\mathbf{x})$ to be the signed distance function as described by equation (18);

Step 2: Advance the level set function by solving the convective transport equation (19) for one time step using either SL-IRBFN or Taylor-IRBFN schemes presented in sections §4 and §5, respectively;

Step 3: Re-initialize the level set function that has just been calculated in Step 2 and do the mass correction;

Step 4: The interface as the zero contour of the level set function has now been advanced one time step. Go back to step 2 for further evolution of the moving interface until the predefined time is reached.

6.1 Initialization

At time $t = 0$, the signed distance function in (18) is defined as the distance from the given collocation point \mathbf{x} to the initial interface curve and the sign is chosen to be positive if the point is inside the curve, and negative if outside,

$$d(x_i, y_i, 0) = \pm \min \|\mathbf{x} - \mathbf{x}_i\|, \mathbf{x}_i \in \Gamma_0, \quad (56)$$

where $\Gamma_0 = \Gamma(0)$ is the initial interface whose discrete representation is \mathbf{x}_i .

6.2 Advancing the level set function

The procedure for advancing the level set function with time by the SL-IRBFN scheme presented in section §4 consists of the following steps

Given \mathbf{v}^0 , for any $\mathbf{x} \in \Omega$ and $n = 0, 1, \dots, N$

1. Compute the departure points \mathbf{x}_d at $t = t^n$ corresponding to the grid point $\mathbf{x} = \mathbf{x}_a$ at $t = t^{n+1}$ in (23) and (24) using the semi-discrete IRBFN-based scheme with Runge-Kutta method as described in [Mai-Cao and Tran-Cong (2005)];
2. Calculate ϕ_d^n at the departure points \mathbf{x}_d by interpolating the known values of $\phi(\mathbf{x}, t^n)$ at the grid points using the IRBFN method;
3. Advance $\phi(\mathbf{x}, t)$ one time step by assigning $\phi^{n+1} = \phi_d^n$

6.3 Calculation of ϕ at departure points

As mentioned earlier, since the departure points \mathbf{x}_d do not coincide with the grid points, the values of ϕ_d^n at those points are obtained by interpolation. The IRBFN formulation is used for this purpose as follows.

$$\phi(\mathbf{x}, t) = \mathbf{g}^T(\mathbf{x}) \mathbf{G}^{-1} \boldsymbol{\phi}(t) \quad (57)$$

where $\boldsymbol{\phi}(t)$ is the values of $\phi(\mathbf{x}, t)$ at all data points \mathbf{x} at time t . The values of ϕ_d^n at the departure points \mathbf{x}_d are obtained by IRBFN interpolation as follows.

$$\phi(\mathbf{x} = \mathbf{x}_d, t = t^n) = \mathbf{g}^T(\mathbf{x} = \mathbf{x}_d) \mathbf{G}^{-1} \boldsymbol{\phi}(t = t^n). \quad (58)$$

It is noted that \mathbf{G}^{-1} needs to be calculated only once, and thus only matrix-vector operations are performed at each time step for interpolation.

6.4 Re-initialization and mass correction

The reinitialization step is done by solving equation (20) to steady-state using the semi-implicit IRBFN-based scheme with the fourth-order Runge-Kutta procedure [Mai-Cao and Tran-Cong (2005)]. The mass correction is then performed to ensure mass conservation as follows. Suppose that after advancing the level set function at time step $t = t^{n+1}$, one gets the moving interface Γ that bounds the domain $\Omega_2 = \mathbf{x} \in \Omega : \phi < 0$. To correct the area of Ω_2 , one changes the zero level set to certain neighboring isoline based on the fact that it has almost the same shape since ϕ is a distance function. This can be done by simply moving the level set function upward or downward by an amount of c_ϕ , where $|c_\phi|$ is the distance between the old and the new zero-level sets

$$\phi^{new} = \phi - c_\phi, \quad (59)$$

where ϕ^{new} is the new (raised or lowered) level set function, $\Omega_2^{new} = \{\mathbf{x} \in \Omega : \phi^{new} < 0\}$. The well-known formula for the first variation of a volume integral [Cuvelier and Schulkes (1990)] is then used to calculate c_ϕ as follows.

$$S_{exact} - S(\Omega) = \int_{\Omega_2^{new}} d\Omega - \int_{\Omega_2} d\Omega = \int_{\Gamma} (c_\phi \mathbf{n}) \cdot \mathbf{n} d\Gamma + \mathcal{O}(c_\phi^2), \quad (60)$$

or

$$S_{exact} - S(\Omega_2) = c_\phi \int_{\Gamma} d\Gamma + \mathcal{O}(c_\phi^2), \quad (61)$$

where S_{exact} is the given exact area of the region, and $S(\Omega_2)$ is the area of Ω_2 . It follows that

$$c_\phi = \frac{S_{exact} - S(\Omega_2)}{L(\Gamma)}, \quad (62)$$

in which $L(\Gamma)$ is the length of the interface Γ . It is noted from equation (62) that if $S_{exact} > S(\Omega_2)$ then $c_\phi > 0$, and the level set function ϕ is to be lowered, meaning that the domain Ω_2 expands. Otherwise, the domain shrinks. In both cases, the level set function is corrected accordingly. In this way, the reinitialization procedure prevents an accumulation of numerical errors in a long run as shown in the numerical results.

7 Numerical results

Some numerical tests are performed in this section to verify the individual numerical schemes as well as the new meshless approach presented in the previous sections. The first test is for checking the capability of the SL-IRBFN and Taylor-IRBFN schemes in dealing with shock wave propagation. The next two problems provide basic tests on the accuracy and efficiency for the new meshless approach to capturing moving interfaces of a solid circle that translates and rotates in a cavity. The present approach is then demonstrated with the simulation of more complicated passive transport problems in which bubbles are moving, stretching and merging together in a divergence-free shear flow.

7.1 Test 1 - Convective transport problems

Test problem 1.1

Consider the propagation of a cosine profile governed by the following convective transport equation

$$\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} = 0, \quad x \in \Omega \quad (63)$$

with the following initial condition

$$u(x, 0) = \begin{cases} \frac{1}{2}(1 + \cos(\pi(x - x_0)/\sigma)) & |x - x_0| \leq \sigma \\ 0 & \text{otherwise} \end{cases} \quad (64)$$

where $c = 1$ is the propagation speed.

The steep profile of the solution is well captured by the SL-IRBFN scheme as shown in Figure 1. In fact, with $N = 61$ regularly located points and $CFL = 0.5$, the numerical solutions are accurate up to 3 digits after the decimal point in the steep region whereas the absolute errors by the scheme can be of order 10^{-5} in the flat regions. In addition, it can be seen in Figure 1 that there are no severe errors found right before and after the shock as observed in Lax-Wendroff and second-order Taylor-Galerkin schemes [Donea and Huerta (2003)].

A comparison of accuracy and stability of the TCN-IRBFN scheme using MQ and TPS basis functions is shown in Figure 2. As can be seen from the figure, the MQ-based scheme is more accurate and stable than its counterpart TPS-based scheme. For this test, the β parameter of MQ-RBF is set to 1.0. Figure 3 shows a comparison of accuracy and stability of the TE-IRBFN schemes using MQ and TPS basis functions. It is observed from this test that TE-IRBFN scheme using MQ-RBF again yields better solution in terms of both accuracy and stability than its TPS-based counterpart.

7.1.1 Test problem 1.2

Consider a convective transport equation

$$\frac{\partial u}{\partial t} - (\sin x) \frac{\partial u}{\partial x} = 0, \quad x \in \Omega, \quad t \in (0, \frac{\pi}{2}], \quad (65)$$

subject to the initial condition

$$u(x, 0) = \sin x \quad (66)$$

The analytical solution to the problem is

$$u(x, t) = \sin \left(2 \tan^{-1} \left(\exp^t \tan \frac{x}{2} \right) \right), \quad (67)$$

which develops sharp layers near the end points $x = 0$ and $x = 2\pi$. The problem is solved up to time $t = \pi/2$ by the Taylor-IRBFN scheme using regularly located points with point spacing as large as $1/10$. Figure 4 shows the analytical and numerical solution by the TCN-IRBFN scheme. As can be seen from the figure, sharp layers near the end points are well resolved by the present scheme even with rather large time steps ($\Delta t = 0.1$ or $CFL=1$). In this case, the time-step size depends on the accuracy requirement, not on stability.

For the purpose of investigating the effect of CFL numbers on the accuracy and stability of the new numerical schemes, the test problem is solved by the SL-IRBFN and TE-IRBFN schemes using a set of different CFL numbers. The root mean square errors corresponding to the CFL numbers are calculated as follows.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{nt} (u_n - u_e)^2}{nt}}, \quad (68)$$

where u_n and u_e are the numerical and exact solutions, respectively; nt is the total number of time steps. As can be seen in Figure 5, for various CFL numbers widely ranging from 0.5 to 4, the root mean square errors are bounded to $\mathcal{O}(10^{-3})$ for the SL-IRBFN scheme, and $\mathcal{O}(10^{-4})$ for the TE-IRBFN scheme. This verifies the accuracy and stability of the two new numerical schemes. It is observed from the test that on the one hand, the TE-IRBFN scheme is not sensitive to CFL number, meaning that the scheme works fine with high CFL numbers. On the other hand, it is also noted that unlike the SL-IRBFN scheme where the value of the unknown at each time step can be found explicitly, the Taylor-IRBFN scheme requires a solution of a system of equations at each time step.

7.2 Test 2 - Solid body translation

In this test problem, a circle of radius 0.5, initially centered at $(-0.75, 0)$, translates to the right due to an external velocity field $\mathbf{v} = (u, v) = (1, 0)$. The objective of the test is to check the accuracy and stability of the new meshless approach in capturing the moving interface. The circle is translated until time $t = 1.0$, and the percentage change in the area is calculated for verification purpose.

The problem is solved by the present meshless approach with uniform point spacing $dx = 1/15$ and time-step size $dt = 0.0667$. The level set function is advanced in time by the Taylor-IRBFN scheme.

Figure 6 shows zero contours of the level set function at different points in time by the TCN-IRBFN scheme. At each time step of interest, the zero contour of the level set function which is the moving interface is extracted using standard contouring algorithm, and the corresponding area of the circle at those time steps are calculated and compared to the exact area of the original circle. As can be seen from the figure, the circle is well captured by the present approach at different points in time.

Figure 7 shows the percentage change in area at different points in time of interest. It can be seen from the figure that the present approach with the TCN-IRBFN scheme is not only able to accurately capture the moving interface but also stable with the error bounded within $\mathcal{O}(10^{-5})$ over the computational time domain. The percentage changes in area at different points in time in this test show that with a coarser point density ($dx = 1/15$) and a larger time step ($dt = 0.0667$), the present meshless approach (using either TE-IRBFN or TCN-IRBFN scheme for solving the level set function) gives more accurate solutions ($\%error \sim \mathcal{O}(10^{-5}) - \mathcal{O}(10^{-3})$) than those resulted from the mesh-based level set scheme ($\%error \sim \mathcal{O}(10^{-2})$) with denser discretization in space ($dx = 1/80$) and time ($CFL = 0.9$) [Sethian (1999)]. Figure 7 also shows that the TCN-IRBFN scheme yields better result than the TE-IRBFN scheme for this test problem.

It is noted that for such a simple velocity field in this test, the reinitialization step is not needed. In fact, only a mass correction presented in Section §6.4 is performed at each time step in this test. Without the reinitialization step, the present approach is still highly accurate and stable. This verifies the efficiency of the new meshless approach for this basic test problem.

7.3 Test 3 - Rotation of a solid body

Consider the rotation of a circle of radius $r = 0.5$ initially centered at $(-0.75, 0)$ in a vortex flow with velocity field $(u, v) = (-y, x)$. It is noted that with such a velocity field, the circle rotates around the coordinate's origin $(0, 0)$ without any deformations. In other words, the circle is considered to be a solid body. An half cycle of rotation is performed by the present meshless approach, and the percentage change in area of the circle during its motion is calculated.

The problem is solved by the present meshless approach with point density $dx = dy = 1/12$ and time-step size $dt = \pi/100 = 0.0314$. The level set function is advanced in time by the SL-IRBFN

scheme (described in Section §4 in which the IRBFN semi-discrete scheme [Mai-Cao and Tran-Cong (2005)] with the fourth-order Runge-Kutta procedure is used to track particles that arrive at the grid points backward to their departure points over a single time step. The function values at those departure points are then obtained by interpolation with TPS-IRBFN formulation.

Figure 8 shows the zero contours of the level set function at different points in time. Although using a rather coarse point density and normal time-step size, the present approach still exactly reconstructs the moving circle at the points in time of interest. Figure 9 presents the percentage errors in area at different points in time. With a very coarse point density ($dx = 1/12$) and a large time-step size ($dt = 0.0314$), the present meshless approach, using the SL-IRBFN scheme for solving the level set function, gives the solution after an half cycle of rotation with the change/error in area of 0.006970%. In [Sethian (1999)], the same test problem was performed with different grid sizes and the percentage error in area was reported to be 0.09758% with the grid size of 161×161 . No conclusion on which (meshless or mesh-based scheme) is better is made for this particular test problem since there is no information about the time-step size used in [Sethian (1999)].

It is noted that the numerical solution for this test is obtained without the reinitialization step. In fact, only a mass correction step is performed at each time step. The numerical result shows that the present approach is accurate and stable with the error bounded within $\mathcal{O}(10^{-4}) - \mathcal{O}(10^{-3})$. It can be concluded that for such a simple velocity field like the one in this test or in Test 2, the reinitialization step is not required provided that a mass correction is performed at each time step. Since no PDEs are solved in the mass correction procedure, saving of computational time is achieved.

7.4 Test 4 - Passive transport of a bubble in a shear flow

In this problem (this example and the following one represent more serious application of the present approach), a bubble with a radius of 0.15, initially centered at (0.5, 0.7) moves and deforms in a shear flow with a divergence-free velocity field $\mathbf{v} = (u, v)$ defined as follows.

$$u = -\sin \pi x \cos \pi y, \quad 0 \leq x, y \leq 1, \quad t \geq 0, \quad (69)$$

$$v = \cos \pi x \sin \pi y, \quad 0 \leq x, y \leq 1, \quad t \geq 0. \quad (70)$$

The problem is solved by the present meshless approach with a point density $dx = dy = 1/50$ and time-step size $dt = 0.01$. The time-step size dt is so chosen to satisfy the Courant-Friedrichs-Levy condition [Osher and Fedkiw (2003)].

$$\Delta t \times \max \left\{ \frac{|u|}{dx} + \frac{|v|}{dy} \right\} = \text{CFL}, \quad (71)$$

where the CFL number is chosen to be unity.

In this problem, the level set function is advanced in time by the SL-IRBFN scheme in which the values of the level set function at departure points are obtained via interpolation by the TPS-IRBFN formulation. At the end of each time step, the reinitialization procedure is done by solving equation (20) to steady-state using the semi-implicit IRBFN-based scheme with the fourth-order Runge-Kutta procedure [Mai-Cao and Tran-Cong (2005)]. For the purpose of investigating the effect of the mass correction on the accuracy and stability of the present approach, the reinitialization procedure is done with and without mass correction. The area of the bubble in motion is calculated at each time step and compared to the original area (πR^2). The error in area of the bubble in motion throughout the simulation time is then used to check the stability of the present approach.

Figures 10-13 show the zero contours (left) and level set function (right) at different points in time.

Figure 14 shows a comparison of the percentage error in area of the bubble resulted from the present approach with and without mass correction for this problem. As can be seen from the figure, the accuracy of the numerical solution is improved significantly with mass correction. In addition, the percentage error of the bubble is bounded within $\mathcal{O}(10^{-5})$ - $\mathcal{O}(10^{-4})$, indicating the good stability of the present approach with mass correction.

7.5 Test 5 - Passive transport of four bubbles in a shear flow

The purpose of this example is to demonstrate the ability of the new meshless approach in dealing with topological changes of interfaces in passive transport problems. Four bubbles, each having a

radius of $R = 1/6$, are initially centered as shown in the top of Figure 15. The bubbles move in a domain of $(-1, 1) \times (-1, 1)$ where there exists a shear flow with the velocity field defined as follows.

$$u = -y \frac{\max \left\{ 1 - (1 - x^2 - y^2)^4, 0 \right\}}{8(x^2 + y^2)} \quad (72)$$

$$v = x \frac{\max \left\{ 1 - (1 - x^2 - y^2)^4, 0 \right\}}{8(x^2 + y^2)} \quad (73)$$

The problem is solved by the present meshless approach with a uniform point spacing $dx = dy = 1/60$ and time-step size $dt = 0.0678$. The time-step size dt is so chosen to satisfy the Courant-Friedrichs-Levy condition as in the previous example.

In this example, the level set function is advanced in time by the SL-IRBFN scheme and the reinitialization procedure is done at each time step. For the purpose of investigating the effect of the mass correction on the accuracy and stability of the present approach, the reinitialization procedure is done with and without mass correction. The total area of the bubbles in motion is calculated at each time step and compared to the original value ($a_0 = 4\pi R^2$). The error in area of the bubbles in motion throughout the simulation time is then used to check the stability of the present approach.

Figures 15-17 show the zero contours (left) and level set function (right) at different points in time.

Figure 18 shows a comparison of the percentage error in area of the bubble resulted from the present approach with and without mass correction for this example. As can be seen from the figure, the accuracy of the numerical solution is improved significantly with mass correction. In addition, the percentage error of the bubble is bounded within $\mathcal{O}(10^{-5})$ - $\mathcal{O}(10^{-3})$, indicating the good stability of the present approach with mass correction.

8 Concluding Remarks

A new meshless approach to capturing moving interfaces in passive transport problems is presented in this paper where the motion and deformation of the moving interfaces are well captured by a unique procedure even with the presence of topological changes. The present approach brings the highly accurate IRBFN method for spatial discretization, the high-order time stepping methods based on semi-Lagrangian or Taylor series expansions and the level set method together for dealing with moving interfaces in an accurate and efficient manner. In this work, a mass correction procedure is introduced at each time step to improve the accuracy of the interface reconstruction. The procedure can be used with or without a reinitialization step. Numerical experiments, including some basic tests for the new numerical schemes and the simulation of one or more bubbles moving, stretching and merging in ambient shear flows, show the good capability of the present approach for this particular moving interface problem. textbfacknowledgement: T. Tran-Cong was hosted by the Faculty of Geology and Petroleum Engineering, on his sabbatical at Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam. This work is partially supported by the Australian Research Council. These supports are gratefully acknowledged.

References

- Allievi, A. and Bermejo, R. (2000). Finite element modified method of characteristics for the Navier-Stokes equations, *International Journal for Numerical Methods in Fluids* **32**: 439–464.
- Atluri, S. (2004). *The Meshless Local Petrov-Galerkin (MLPG) Method for Domain & BIE Discretizations*, Tech Science Press, Encino, CA, USA.
- Bardi, M. and Osher, S. (1991). The nonconvex multidimensional Riemann problem for Hamilton-Jacobi equations, *SIAM Journal on Mathematical Analysis* **22**: 344–351.
- Behrens, J. and Iske, A. (2002). Grid-free adaptive semi-Lagrangian advection using radial basis functions, *Computers and Mathematics with applications* **43**: 319–327.

- Chang, Y., Hou, T., Merriman, B. and Osher, S. (1996). A level set formulation of Eulerian interface capturing method for incompressible fluid flows, *Journal of Computational Physics* **124**: 449–464.
- Crandall, M. and Lions, P. (1984). Two approximations of solutions of Hamilton-Jacobi equations, *Math. Comput.* **43**: 1–19.
- Cuvelier, C. and Schulkes, R. (1990). Some numerical methods for the computation of capillary free boundaries governed by the Navier-Stokes equations, *SIAM Review* **32**(3): 355–423.
- Donea, J. (1984). A Taylor-Galerkin method for convective transport problems, *International Journal for Numerical Methods in Engineering* **20**: 101–119.
- Donea, J. and Huerta, A. (2003). *Finite Element Methods for Flow Problems*, John Wiley & Sons Ltd., England.
- Floryan, J. and Rasmussen, H. (1989). Numerical methods for viscous flows with moving boundary, *Applied Mechanics Reviews* **42**(12): 323–341.
- Hirt, C. and Nichols, B. (1981). Volume of fluid (VOF) method for the dynamics of free boundaries, *Journal of Computational Physics* **39**(1): 201–225.
- Iafrati, A., Mascio, A. and Campana, E. (2001). A level set technique applied to unsteady free surface flows, *International Journal for Numerical Methods in Fluids* **35**: 281–297.
- Jacqmin, D. (1999). Calculation of two-phase Navier-Stokes flows using phase-field modelling, *Journal of Computational Physics* **155**: 96–127.
- Kansa, E. (1990). Multiquadrics - A scattered data approximation scheme with applications to computational fluid-dynamics-II. Solutions to parabolic, hyperbolic and elliptic partial differential equations, *Computers and Mathematics with Applications* **19**(8/9): 147–161.
- Mai-Cao, L. and Tran-Cong, T. (2005). A meshless IRBFN-based method for transient problems, *CMES:Computer Modeling in Engineering & Sciences* **18**(1): 59–77.
- Mai-Duy, N. (2004). Indirect rbf method with scattered points for numerical solution of pdes, *CMES:Computer Modeling in Engineering & Sciences* **6**: 209–226.
- Mai-Duy, N. and Tran-Cong, T. (2001a). Numerical solution of differential equations using multiquadric radial basis function networks, *Neural Networks* **14**: 185–199.
- Mai-Duy, N. and Tran-Cong, T. (2001b). Numerical solution of navier-stokes equations using multiquadric radial basis function networks, *International Journal for Numerical Methods in Fluids* **37**: 65–86.
- Mai-Duy, N. and Tran-Cong, T. (2003). Approximation of function and its derivatives using radial basis function networks, *Applied Mathematical Modelling* **27**: 197–220.
- Mai-Duy, N., Mai-Cao, L. and Tran-Cong, T. (2007). Computation of transient viscous flows using indirect radial basis function networks, *CMES:Computer Modeling in Engineering & Sciences* **7**(2): 149–171.
- Moody, J. and Darken, C. (1989). Fast learning in networks of locally-tuned processing units, *Neural Computations* **1**: 281–294.
- Oliveira, A. and Baptista, A. (1995). A comparison of integration and interpolation Eulerian-Lagrangian methods, *International Journal for Numerical Methods in Fluids* **21**: 183–204.
- Osher, S. and Fedkiw, R. (2003). *Level Set Methods and Dynamic Implicit Surfaces*, Vol. 153 of *Applied Mathematical Sciences*, Springer, New York.
- Osher, S. and Sethian, J. (1988). Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *Journal of Computational Physics* **79**: 12–49.

- Peng, D., Merriman, B., Osher, S., Zhao, H. and Kang, M. (1999). A PDE-based fast local level set method, *Journal of Computational Physics* **155**: 410–438.
- Sarler, B. (2005). A radial basis function collocation approach in computational fluid dynamics, *CMES:Computer Modeling in Engineering & Sciences* **7**(2): 185–193.
- Sethian, J. (1999). *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*, Cambridge University Press, New York.
- Shu, C. and Osher, S. (1989). Efficient implementation of essentially non-oscillatory shock capturing schemes II, *Journal of Computational Physics* **83**: 32–78.
- Shu, C., Ding, H. and Yeo, K. (2005). Computation of incompressible navier-stokes equations by local rbf-based differential quadrature method, *CMES:Computer Modeling in Engineering & Sciences* **7**(2): 195–205.
- Staniforth, A. and Cote, J. (1991). Semi-lagrangian integration schemes for atmospheric models - A review, *Monthly Weather Review* **119**: 2206–2223.
- Sussman, M. and Fatemi, E. (1999). An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow., *SIAM Journal on Scientific Computing* **20**(4): 1165–1191.
- Sussman, M. and Smereka, P. (1997). Axisymmetric free boundary problems, *Journal of Fluid Mechanics* **341**: 269–294.
- Sussman, M., Smereka, P. and Osher, S. (1994). A level set approach for computing solutions to incompressible two-phase flow, *Journal of Computational Physics* **114**: 146–159.
- Temperton, C. and Staniforth, A. (1987). An efficient two-time level semi-Lagrangian semi-implicit integration scheme, *Quart. J. Roy. Meteor. Soc.* **113**: 1025–1039.
- Tornberg, A. (2000). *Interface Tracking Methods with Application to Multiphase Flows*, PhD thesis, Royal Institute of Technology, Stockholm.
- Unverdi, S. and Tryggvason, G. (1992). A front-tracking method for viscous, incompressible, multi-fluid flows, *Journal of computational physics* **100**: 25–37.
- Wang, S., Lim, K., Khoo, B. and Wang, M. (2007). A geometric deformation constrained level set method for structural shape and topology optimization, *CMES:Computer Modeling in Engineering & Sciences* **18**(3): 155–181.

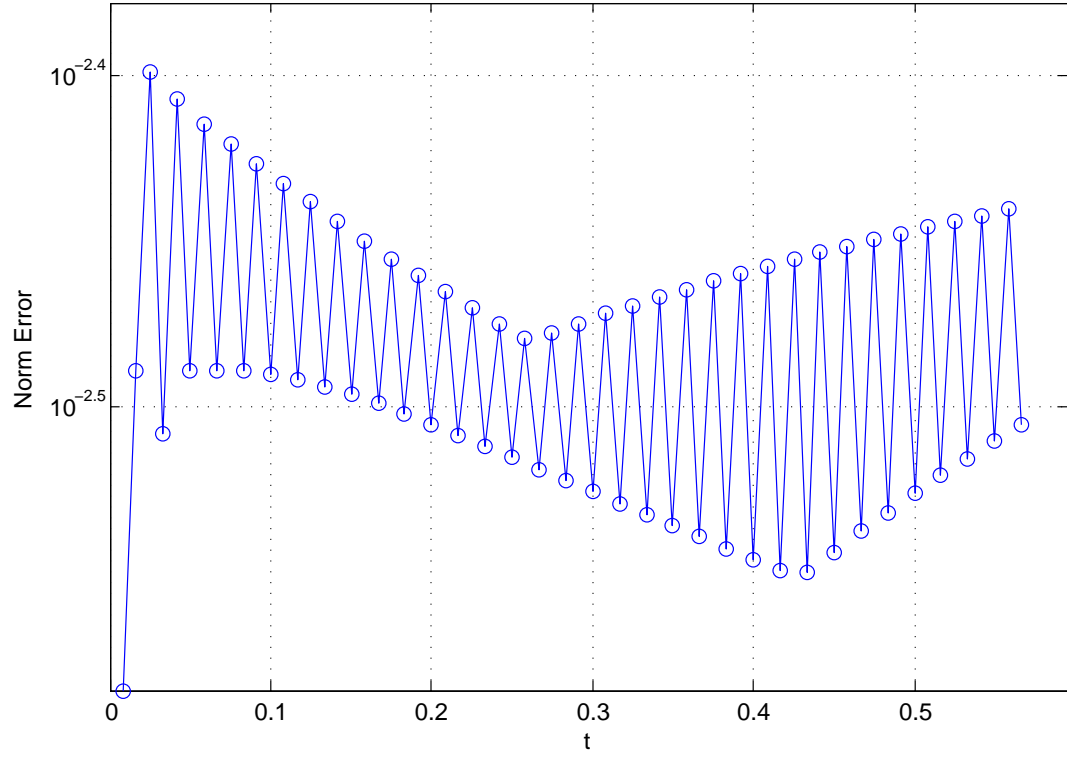
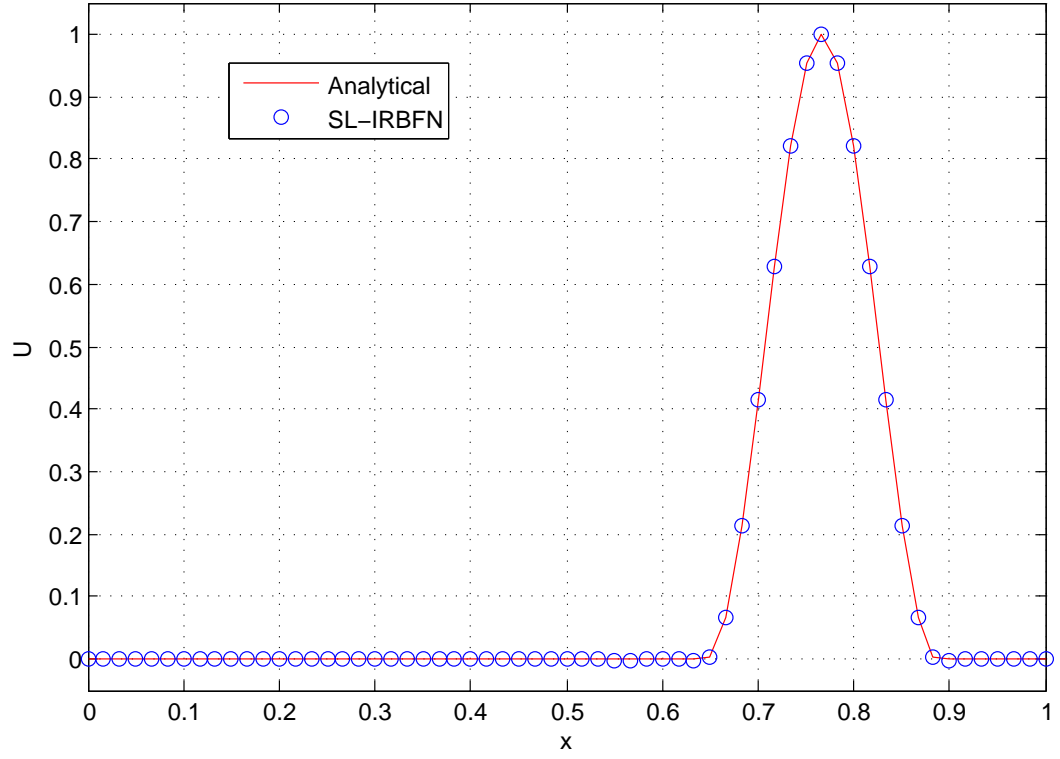


Figure 1: Test 1.1: Numerical solution (top) and its L_∞ -norm error (bottom) by the SL-IRBFN scheme.

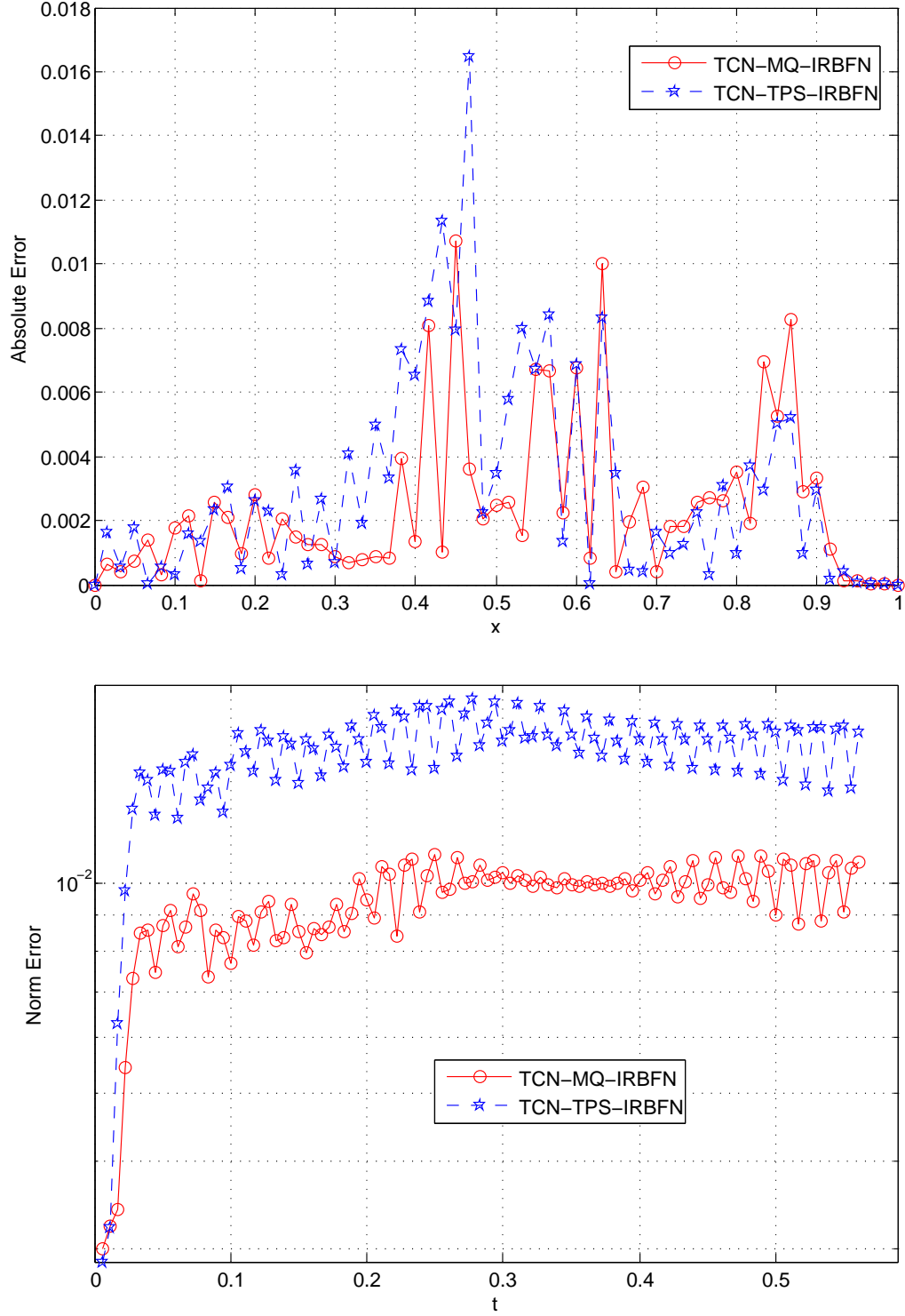


Figure 2: Test 1.1: Comparison between the two variants of the TCN-IRBFN scheme (using MQ and TPS basis functions), (Top) Absolute error at the last time step; (Bottom) L_∞ -norm error with respect to time.

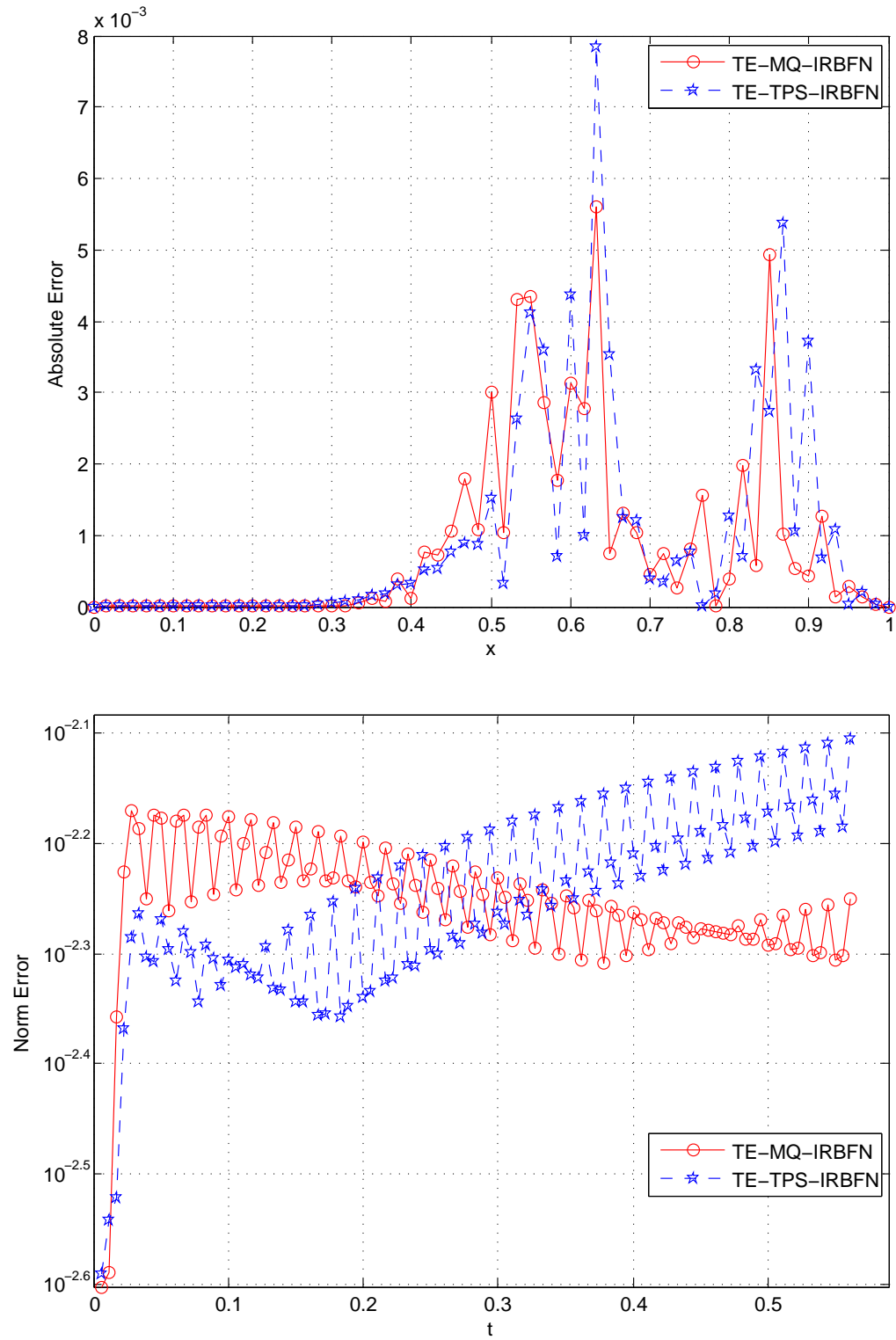


Figure 3: Test 1.1: Comparison between the two variants of the TE-IRBFN scheme (using MQ and TPS basis functions), Absolute error at the last time step (top); L_∞ -norm error with respect to time (bottom).

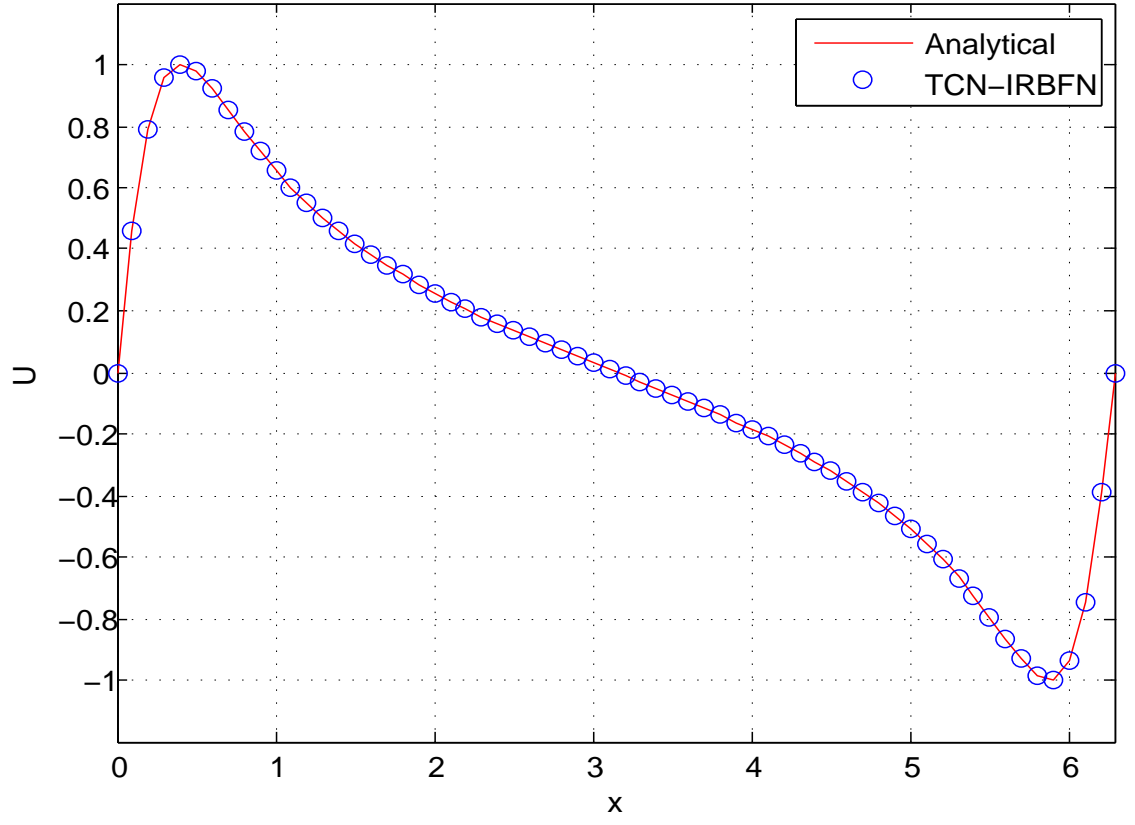


Figure 4: Test 1.2: Numerical solution by the TCN-IRBFN scheme. The analytical (solid line) and numerical solution (“o” symbol) are plotted at $t = \pi/2$ with $\text{CFL}=1$ and $dx = 1/10$.

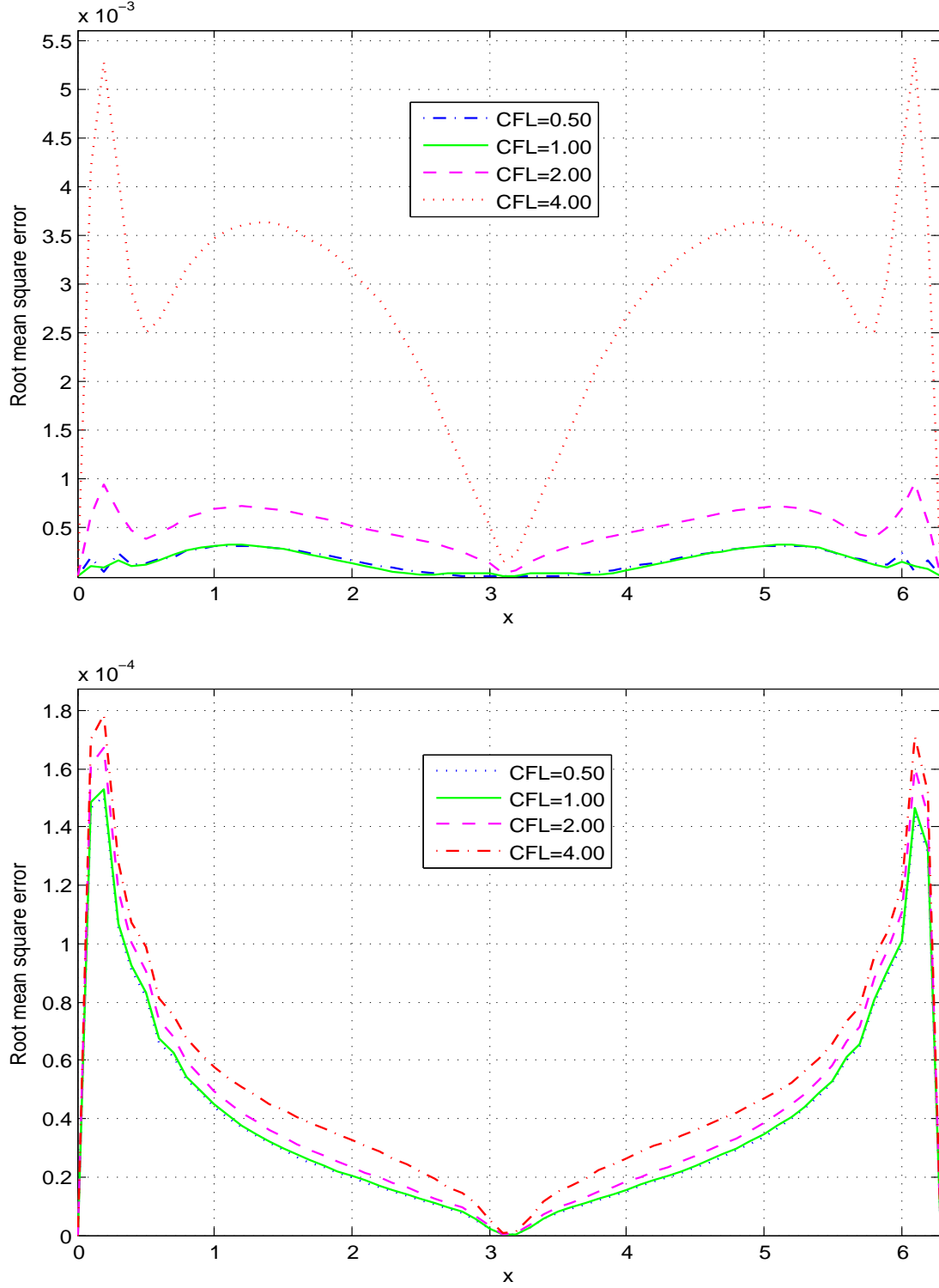


Figure 5: Test 1.2: Numerical investigation of the accuracy and stability of the present numerical schemes, root mean square errors corresponding to various values of the CFL number within a rather wide range (0.5-4) are bounded to $\mathcal{O}(10^{-3})$ for the SL-IRBFN scheme (top), and $\mathcal{O}(10^{-4})$ for the TE-IRBFN scheme (bottom).

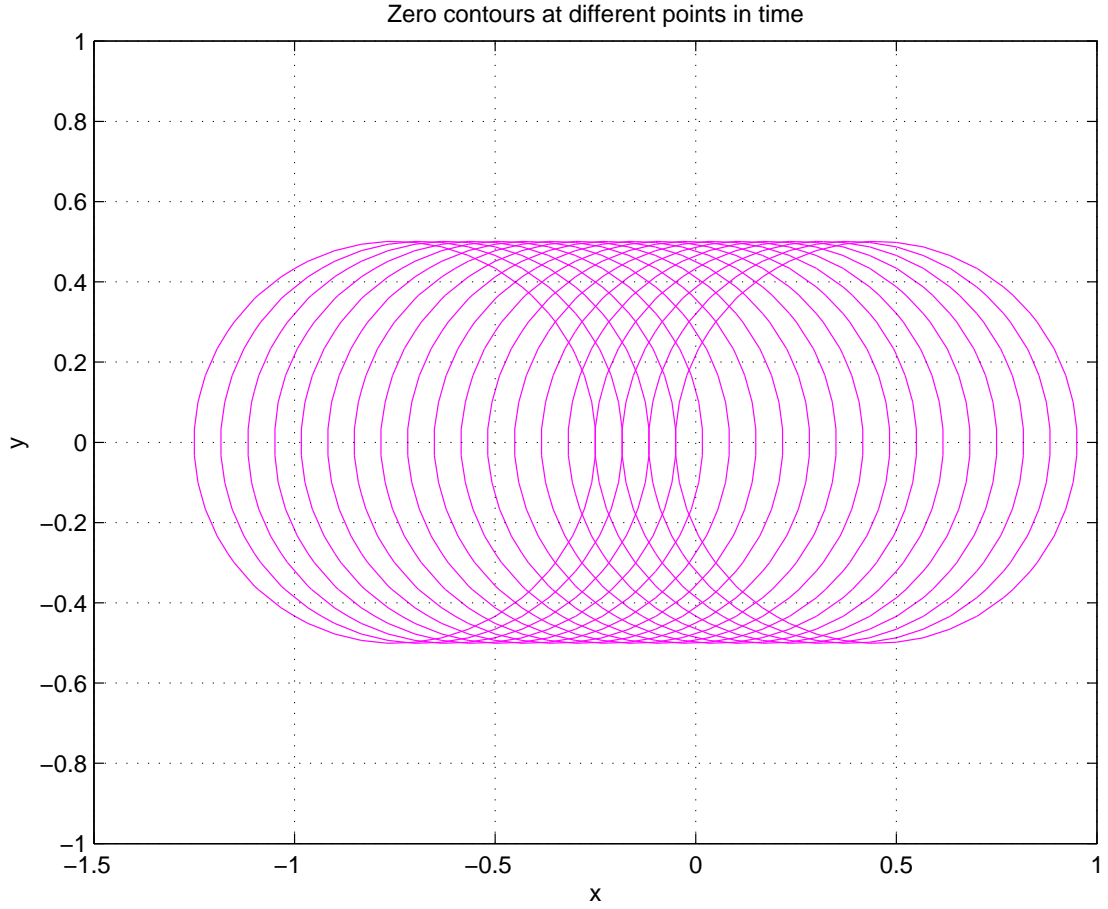


Figure 6: Test 2: Zero contours of the level set function at different points in time by the TCN-IRBFN scheme. Although using a rather coarse point density ($dx = dy = 1/15$) and large time-step size ($dt = 0.0667$), the present approach is still able to exactly reconstruct the moving circle at the points in time of interest.

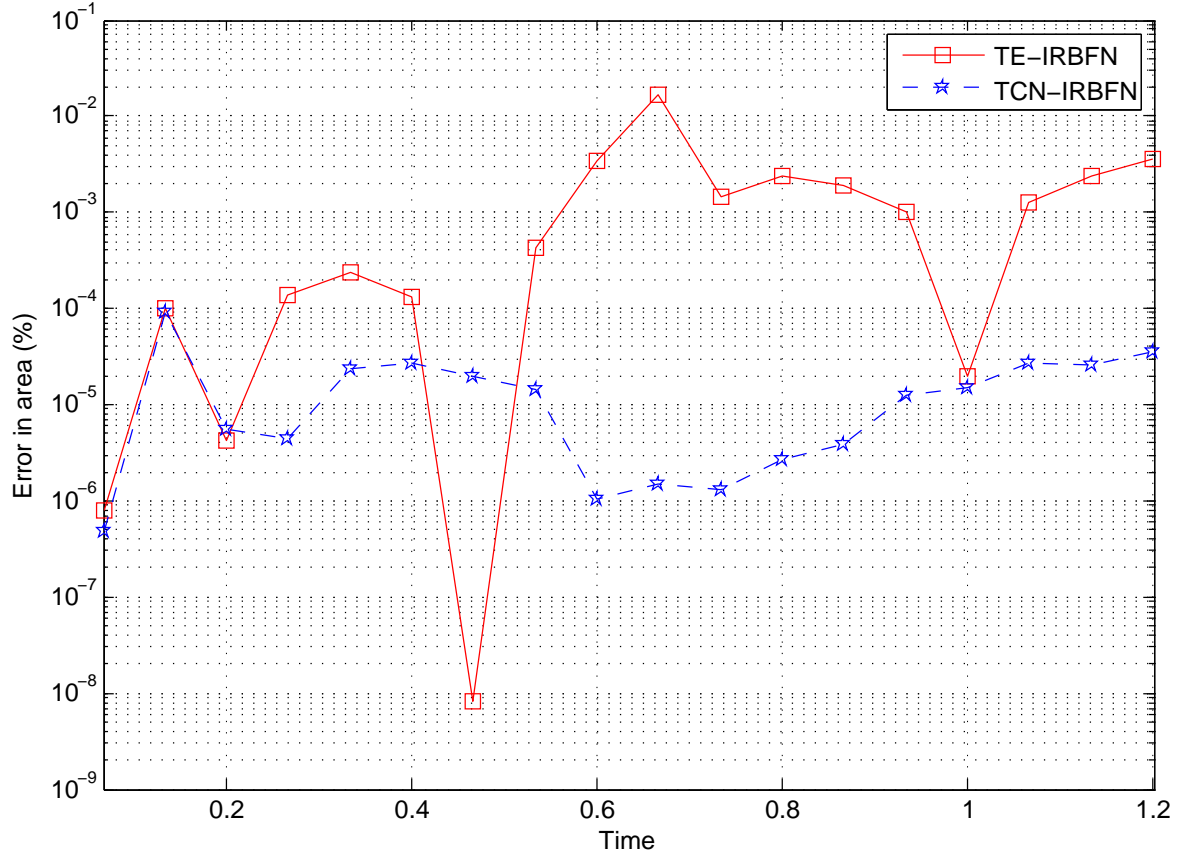


Figure 7: Test 2: Percentage errors in area at different points in time. With a coarser point density ($dx = 1/15$) and a larger time step ($CFL = 1.0$), the present meshless approach (using either TE-IRBFN or TCN-IRBFN scheme for solving the level set function) gives more accurate solutions ($\%error \sim \mathcal{O}(10^{-5}) - \mathcal{O}(10^{-3})$) than those resulted from the mesh-based level set scheme ($\%error \sim \mathcal{O}(10^{-2})$) with denser grid points (81×81) and time ($CFL = 0.9$) [Sethian (1999)]. The numerical result also shows that the TCN-IRBFN scheme is more accurate and stable than the TE-IRBFN scheme for this test problem.

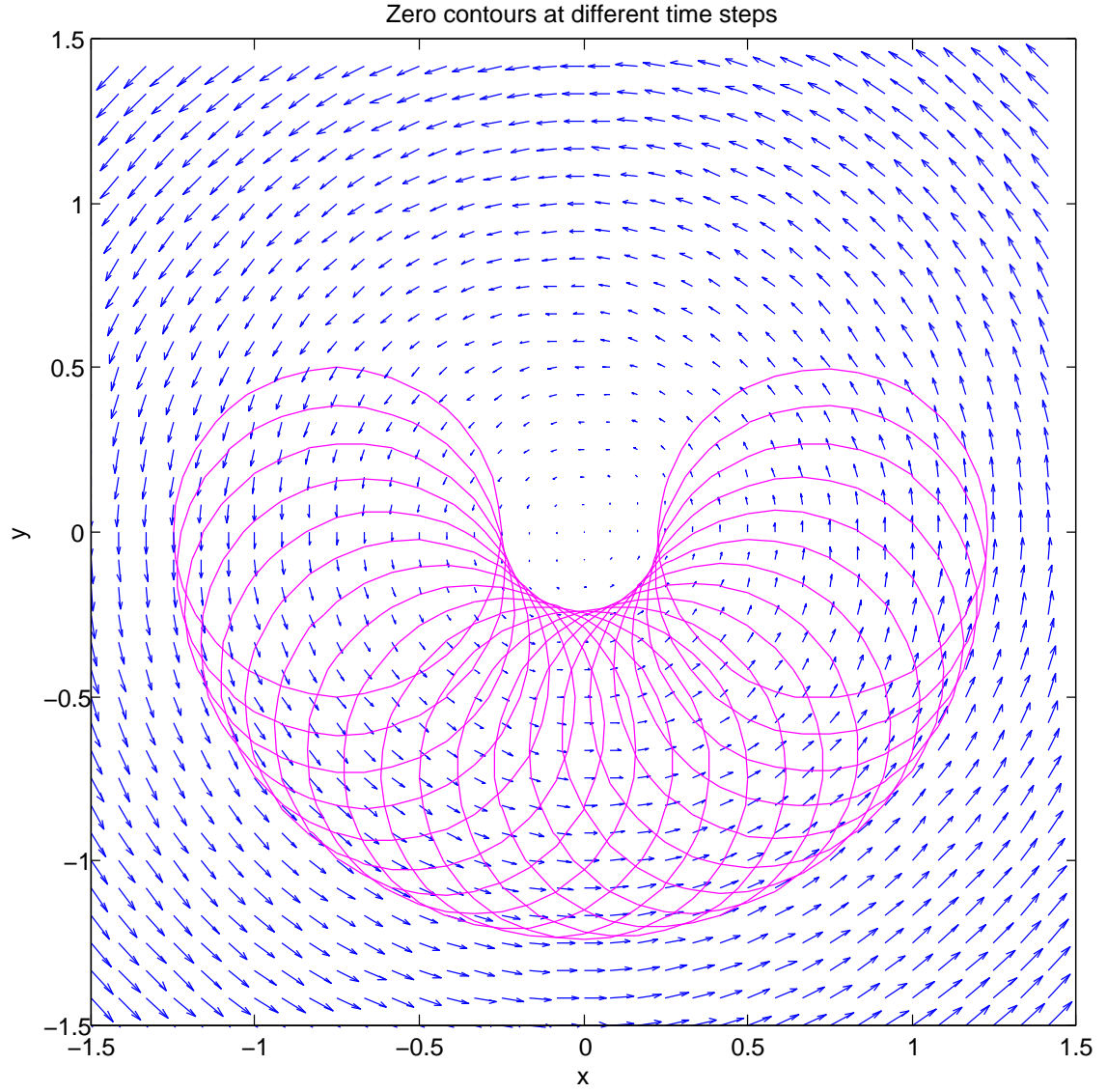


Figure 8: Test 3: Zero contours of the level set function at different points in time. Although using a rather coarse point density ($dx = dy = 1/12$) and normal time step ($dt = 0.0314$), the present approach is able to accurately reconstruct the moving circle at the points in time of interest.

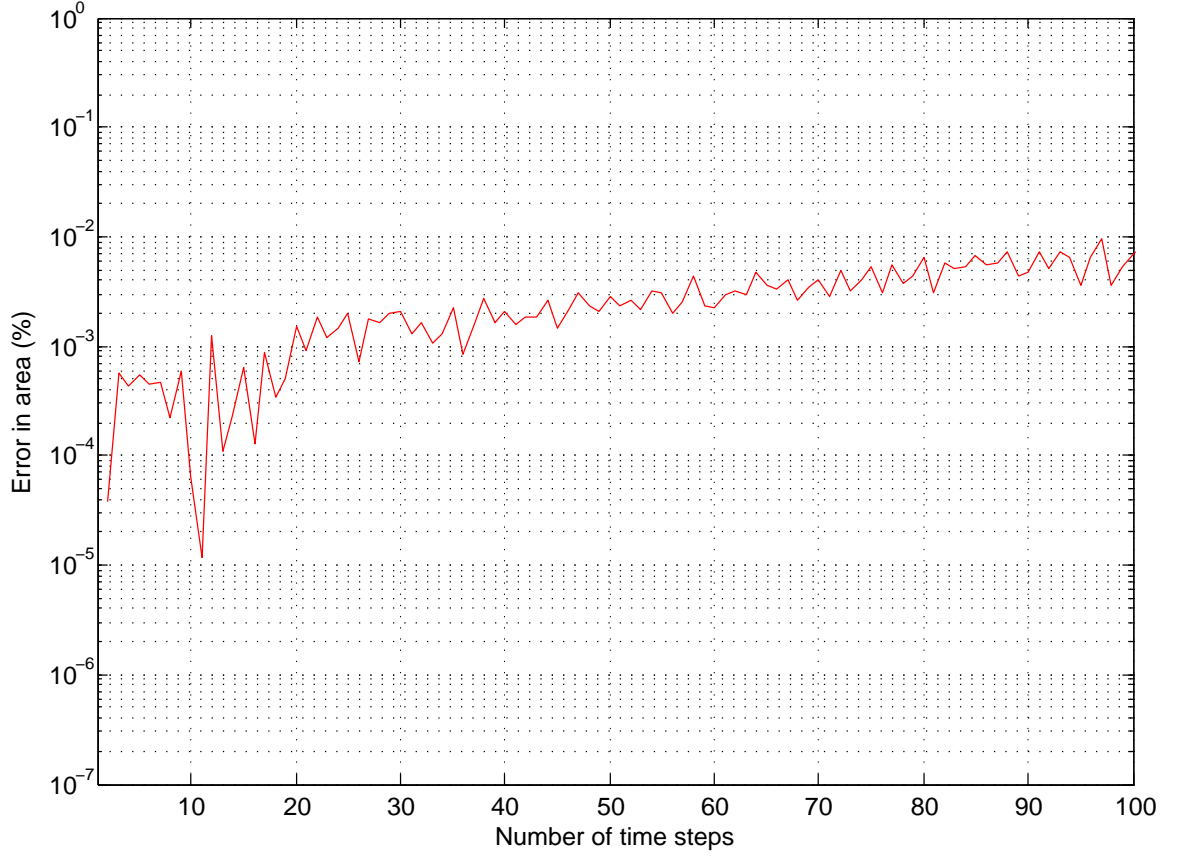


Figure 9: Test 3: Percentage errors in area at different points in time. With a rather coarse point density ($dx = dy = 1/12$) and normal time-step size ($dt = 0.0314$), the present meshless approach, using the SL-IRBFN scheme for advancing the level set function, is quite accurate and stable.

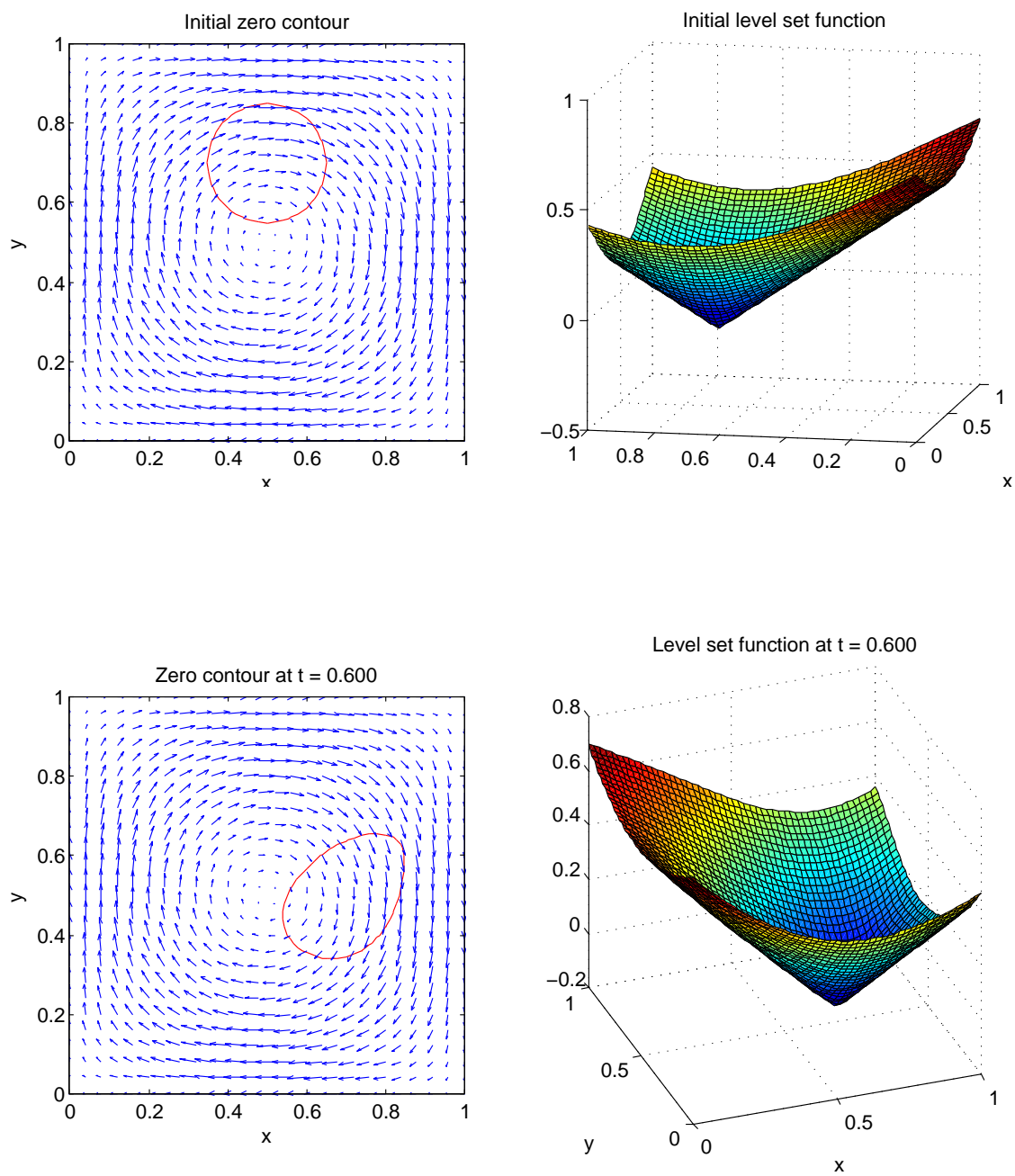


Figure 10: Test 4: The zero contour and the level set function at $t = 0$ (top) and $t = 0.60$ (bottom).

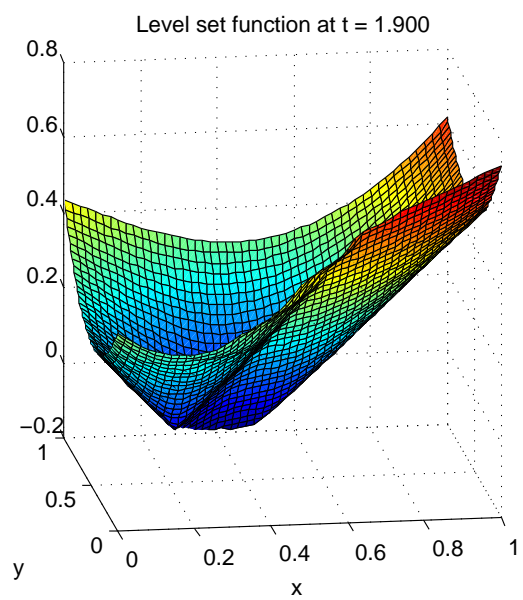
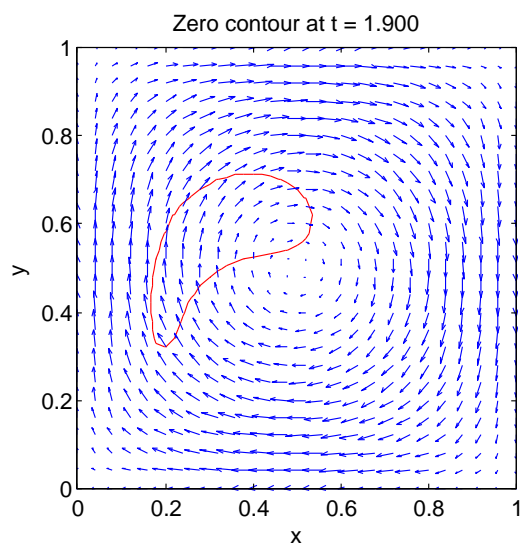
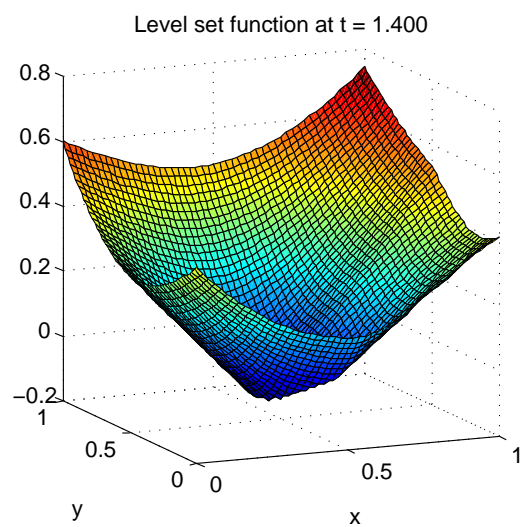
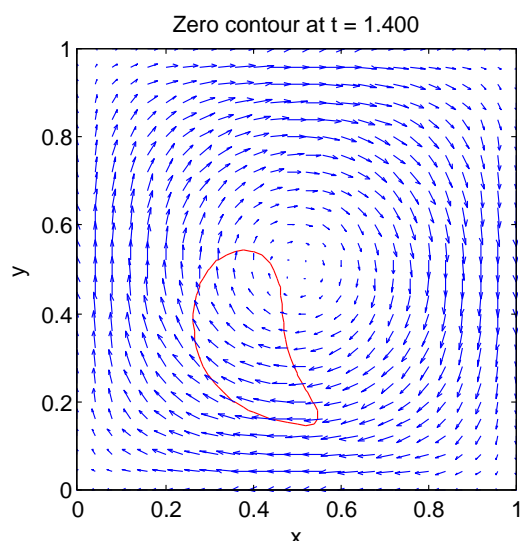


Figure 11: Test 4: The zero contour and the level set function at $t = 1.40$ (top) and $t = 1.90$ (bottom).

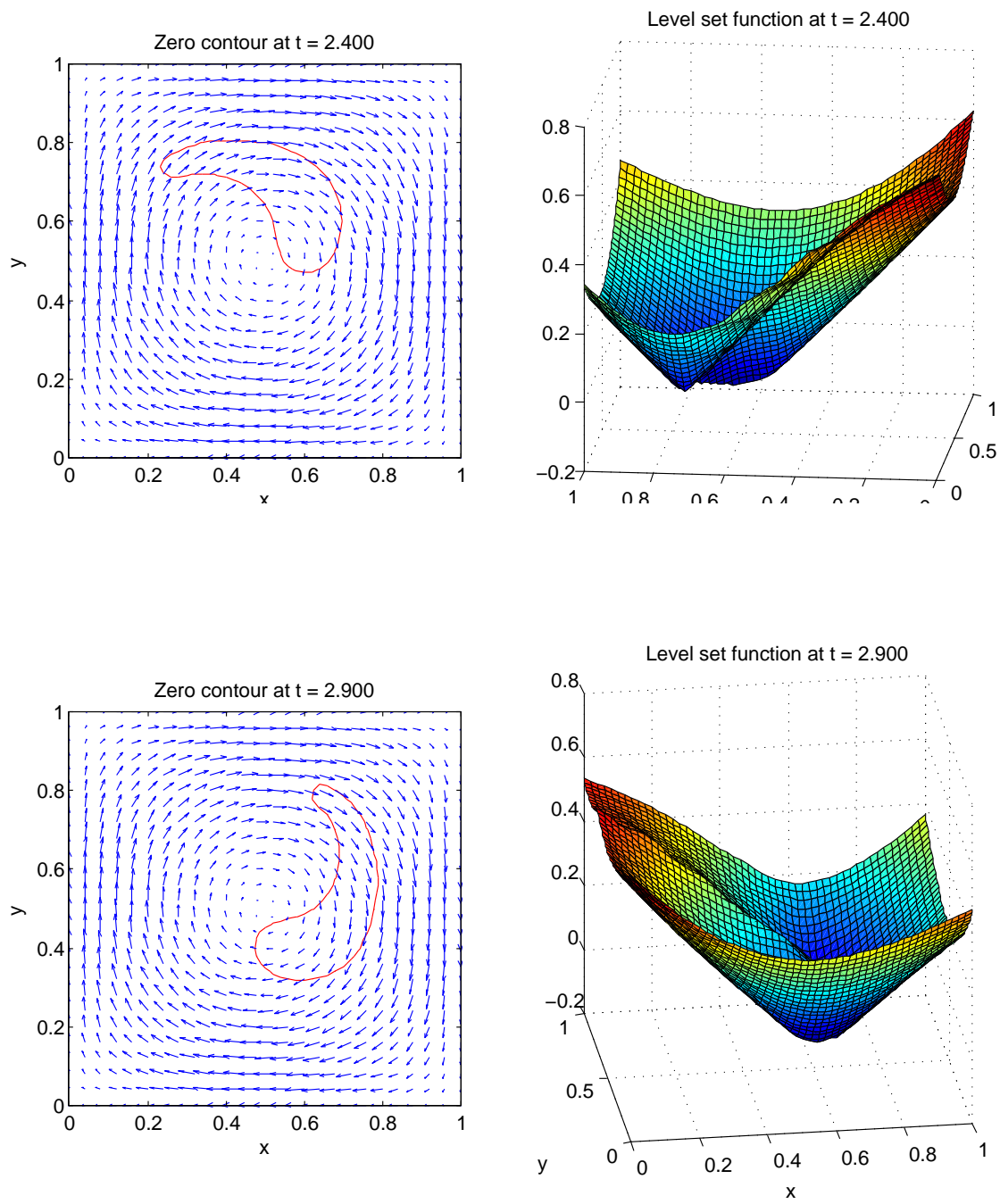


Figure 12: Test 4: The zero contour and the level set function at $t = 2.40$ (top) and $t = 2.90$ (bottom).

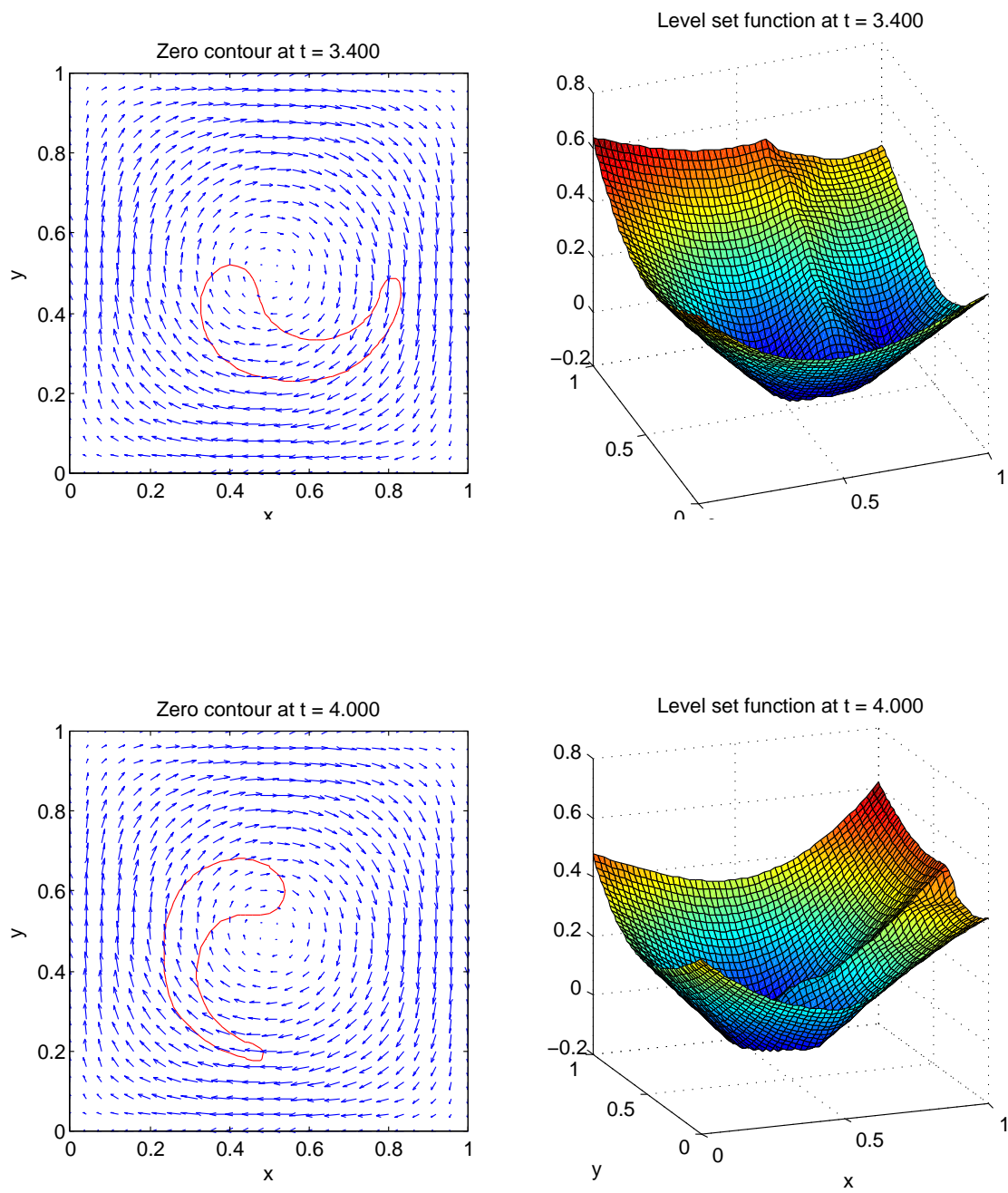


Figure 13: Test 4: The zero contour and the level set function at $t = 3.40$ (top) and $t = 4.00$ (bottom).

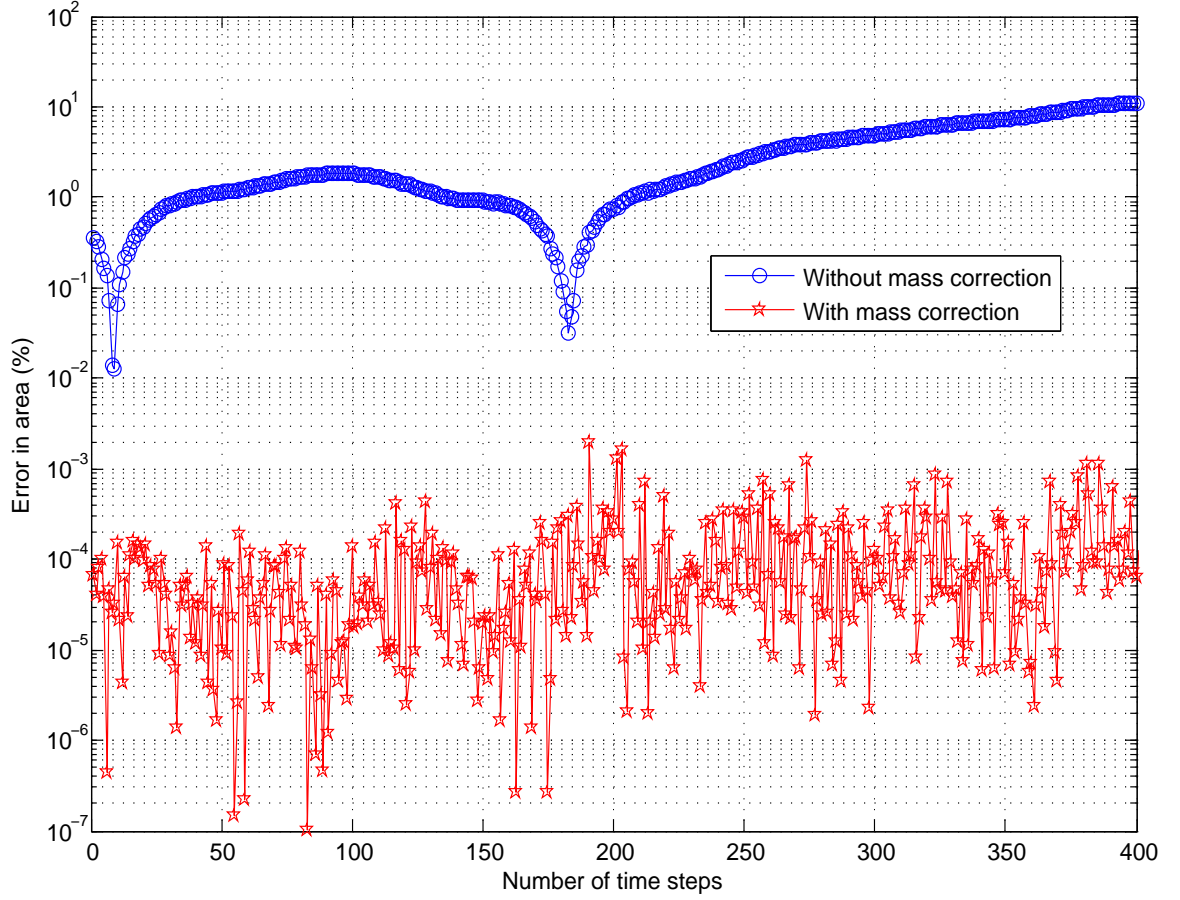


Figure 14: Test 4: Comparison of the percentage error in area of the bubble resulted from the present approach with and without mass correction. The accuracy of the numerical solution is improved significantly with mass correction. The error is bounded within $\mathcal{O}(10^{-5})$ - $\mathcal{O}(10^{-4})$, indicating that the present approach with mass correction is stable for this problem.

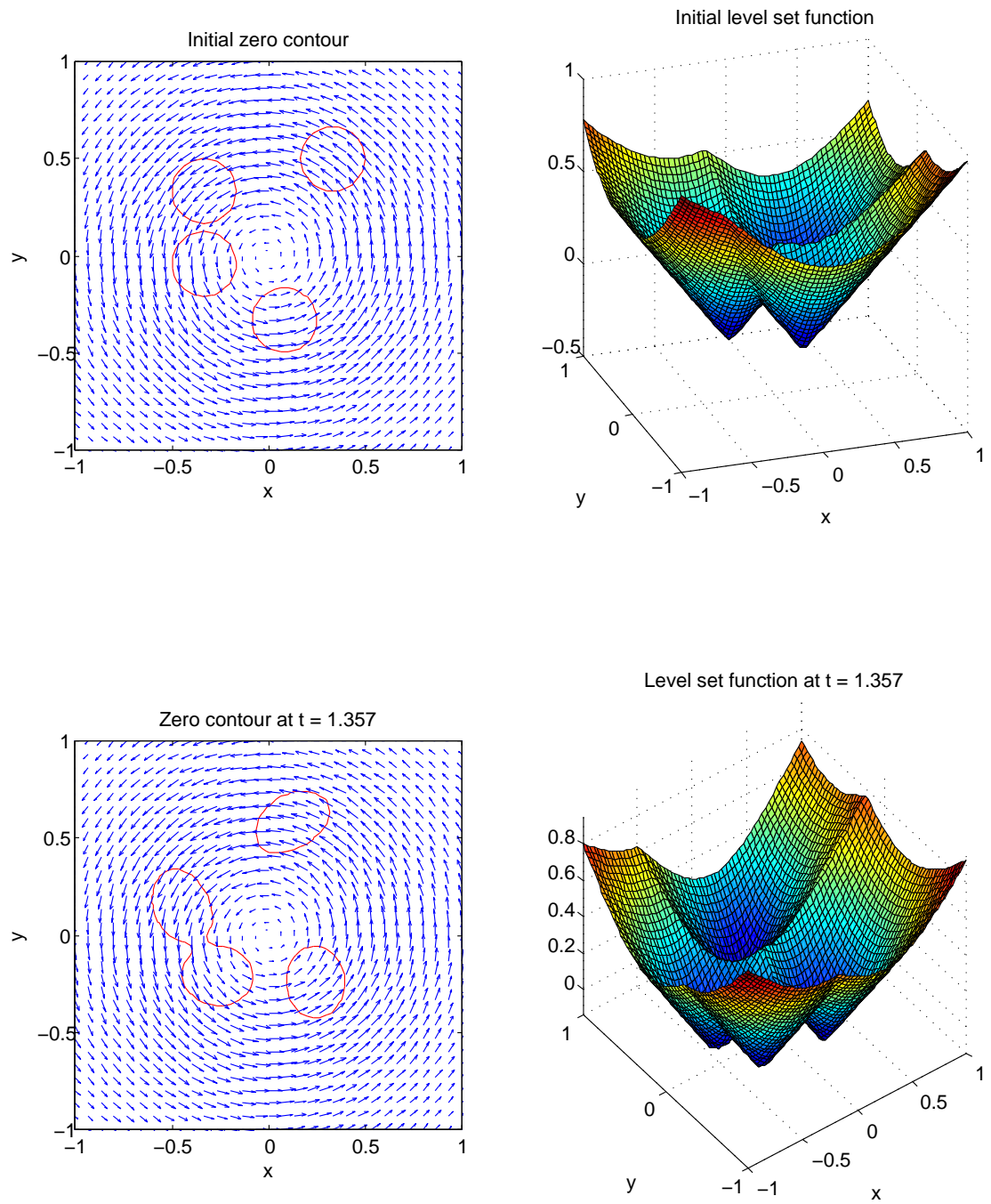


Figure 15: Test 5: Zero contours and the level set function at $t=0$ (top) and $t=1.357$ (bottom).

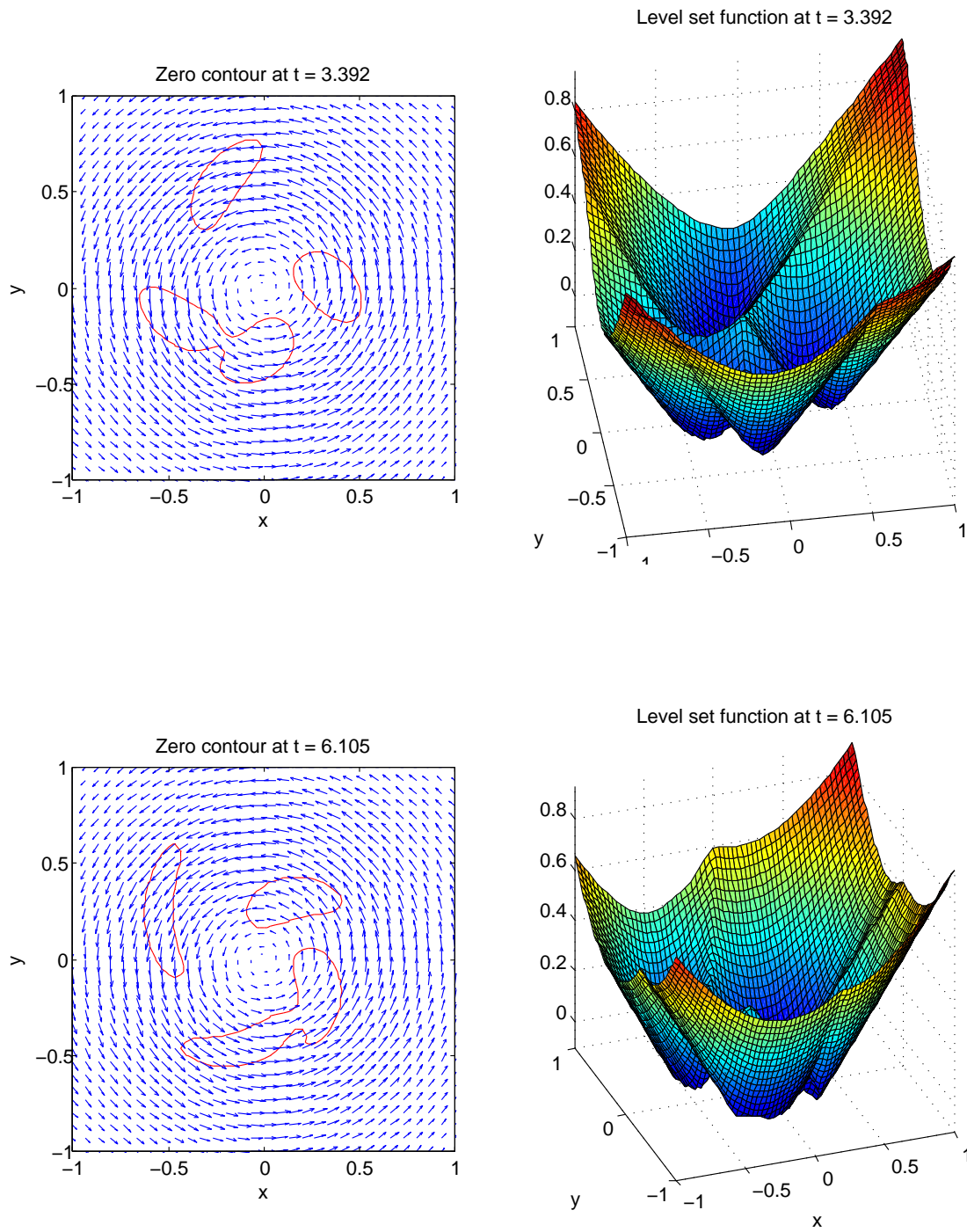


Figure 16: Test 5: Zero contours and the level set function at $t=3.392$ (top) and $t=6.105$ (bottom).

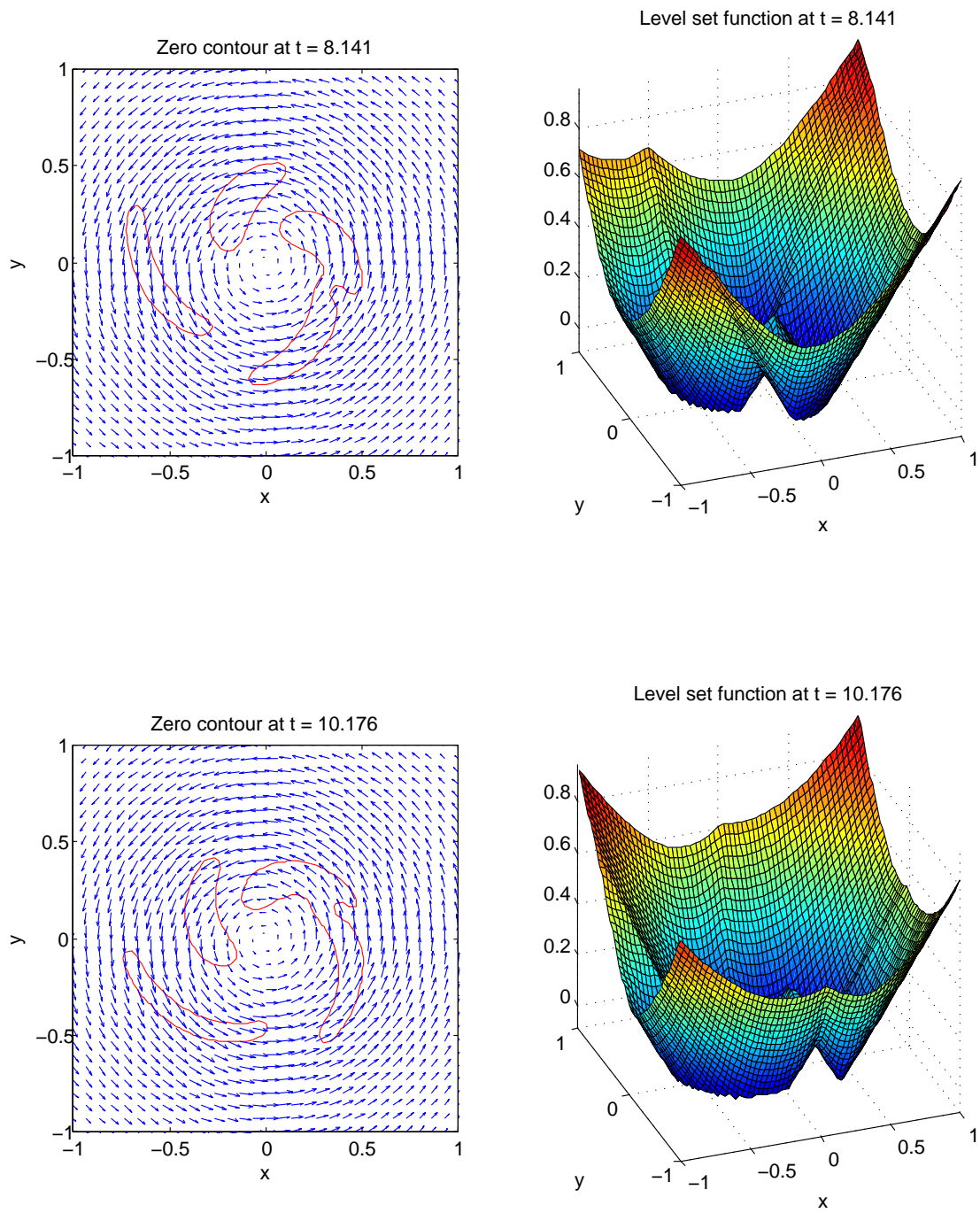


Figure 17: Test 5: Zero contours and the level set function at $t=8.141$ (top) and $t=10.176$ (bottom).

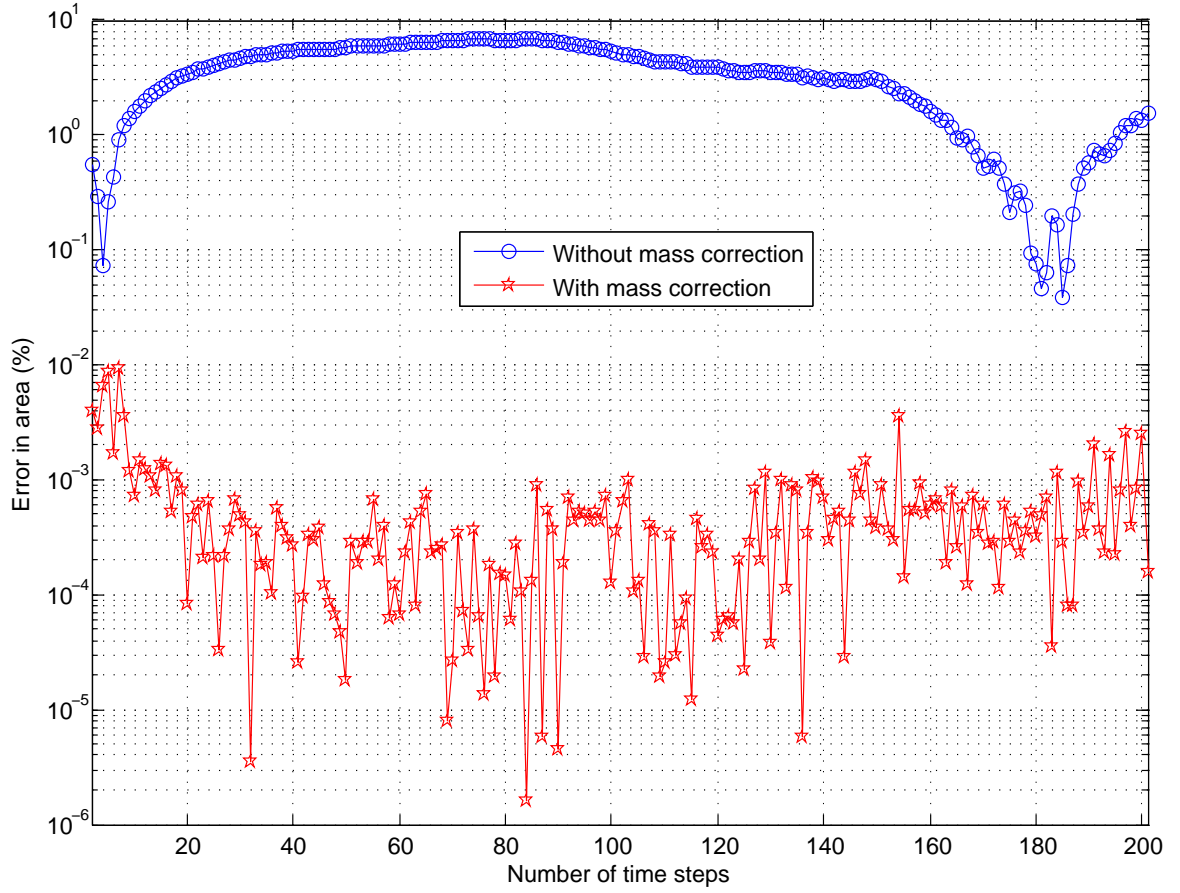


Figure 18: Test 5: Comparison of the percentage error in area of the bubbles resulted from the present approach with and without mass correction. The accuracy of the numerical solution is improved significantly for the latter case. The error in total area of the bubbles in motion is bounded within $\mathcal{O}(10^{-5})$ - $\mathcal{O}(10^{-3})$, indicating the good stability of the present approach with mass correction.